Multi-Root I/O Virtualization and Sharing Specification

Revision 0.9

November 7, 2007

REVISION	REVISION HISTORY	DATE
0.5	Initial release	11/13/2006
0.7	Methods & Functions Complete	6/8/2007
0.9	Final Scrub of 0.9	11/7/2007

PCI-SIG® disclaims all warranties and liability for the use of this document and the information contained herein and assumes no responsibility for any errors that may appear in this document, nor does PCI-SIG make a commitment to update the information contained herein.

Contact the PCI-SIG office to obtain the latest revision of the specification.

Questions regarding this document or membership in PCI-SIG may be forwarded to:

Membership Services
http://www.pcisig.com
E-mail: administration@pcisig.com
Phone: 503-619-0569
Fax: 503-644-6708

Technical Support
techsupp@pcisig.com

DISCLAIMER

Contents

1. ARCHITECTURAL OVERVIEW 15

1.1. How Does MR-IOV Work? 17
1.1.1. MRA Components 22
1.1.2. MR-IOV and ARI (Alternative Routing Identifier) 28
1.1.3. MR-IOV Relationship to SR-IOV and ATS 29
1.2. Overview of MR Transaction Layer 30

2. MR PROTOCOL CHANGES 34

2.1. MR Link and Flow Control Negotiation 34
2.1.1. MR Link Protocol Negotiation 39
2.1.2. MR Flow Control Initialization Protocol 40
2.2. TLP Prefix Tagging 50
2.2.1. MR Switch Transaction Layer Processing 52
2.2.2. MR Device Transaction Layer Processing 53
2.2.3. Global Key Processing 55
2.2.4. MR TLP Dataflow Examples 56
2.3. Per-VH RESET 57
2.3.1. Per-VH Reset Example 58
2.3.2. RESET DLLP Format 61
2.3.3. RESET DLLP Processing 62
2.4. MR Flow Control 67
2.4.1. FC Information Tracked by Transmitter 67
2.4.2. FC Information Tracked by Receiver 70
2.4.3. Electrical Idle Inference 72
2.5. MR Message Processing 73
2.5.1. Interrupts 73
2.5.2. PME Turn Off Processing 74
2.5.3. PM_PME Processing 74
2.6. Miscellaneous Changes 74
2.7. Miscellaneous Non-Changes 74

3. INITIALIZATION AND RESOURCE ALLOCATION 76

3.1. MR Topology Initialization 76
3.1.1. Initial State After Fundamental Reset 77
3.1.2. Initial MR-PCIM Location Policy 81
3.1.3. Topology Discovery 81
3.1.4. Component Discovery 84
3.1.5. VH and VF Mapping Policy 86
3.1.6. VH and VF Mapping Implementation 87
3.1.7. MR-PCIM Failover 94
3.2. MR Device Initialization 95
3.2.1. Enabling MR Operation 96
3.2.2. Managing Flow Control 97

3.2.3. Managing VF Mapping ... 98
3.2.4. Managing VF Migration ... 100
3.3. MR ROOT PORT INITIALIZATION ... 104

4. CONFIGURATION ... 105

4.1. CONFIGURATION FIELD SUMMARY ... 106
4.2. DEVICE CONFIGURATION SPACE ... 112
4.2.1. Device MR-IOV Extended Capability ... 114
4.2.2. Device VL Arbitration Table ... 129
4.2.3. LVF Table ... 130
4.2.4. Function Table ... 131
4.2.5. Misc. Device Configuration Space Requirements ... 146
4.3. SWITCH CONFIGURATION SPACE ... 148
4.3.1. Switch MR-IOV Extended Capability ... 150
4.3.2. Switch VS Authorization Bitmap ... 159
4.3.3. Switch Port Table ... 160
4.3.4. Switch VL Arbitration Table ... 178
4.3.5. Switch VS Table ... 179
4.3.6. Switch VS Bridge Table ... 183
4.3.7. Miscellaneous Switch Configuration Space Requirements ... 198
4.4. VL ARBITRATION TABLE ... 204
4.5. PERFORMANCE MONITORING AND STATISTICS COLLECTION ... 206
4.5.1. Configuration Space Fields ... 208
4.5.2. Statistics Descriptor Table ... 211
4.5.3. Statistics Block Table ... 219
4.5.4. Statistics Counter Table ... 220

ERROR HANDLING ... 224

5. PCIE ERROR MAPPING TO MR ... 224

5.1. MR ERRORS ... 226

6. HOT PLUG ... 230

6.1. MRA SWITCH ... 230
6.1.1. PCI Express Capability: Slot Capability Register ... 231
6.1.2. PCI Express Capability: Slot Control Register ... 232
6.1.3. PCI Express Capability: Slot Status Register ... 234
6.1.4. PCI Express Capability: Device Capabilities Register ... 235
6.1.5. Hot-Plug Virtual Signals Interface Registers ... 236
6.1.6. Physical Slot Registers ... 237
6.1.7. Physical Hot-Plug Signals Interface ... 237
6.2. VIRTUAL DEVICE MIGRATION ... 237
6.3. BASE PCI EXPRESS DEVICE MIGRATION ... 238

7. POWER MANAGEMENT ... 240

7.1. OVERVIEW ... 240
7.2. VIRTUAL D-STATE ... 240

7.3. LINK POWER STATES 240
7.4. MULTI-ROOT ASPM 241
7.5. SLOT CLOCK AND COMMON CLOCK CONFIGURATION 241
7.6. MULTI-ROOT WAKE-UP 242
7.6.1. PME Triggers Beacon/Wake# 242
7.6.2. Beacon/Wake# Triggers MSI 243
7.6.3. Beacon/WAKE# Triggers Beacon/WAKE# 243
7.7. MULTI-ROOT PME TURN OFF 243
7.8. MULTI-ROOT POWER CONTROLLER 245
7.9. MULTI-ROOT POWER BUDGETING 245

8. CONGESTION MANAGEMENT 246

8.1. OVERVIEW 246
8.2. CONGESTION ISOLATION 247
8.2.1. Virtual Links 247
8.2.2. Bypass Queues 253
8.2.3. Flow Control Rules 254
8.3. PERFORMANCE MONITORING AND STATISTICS COLLECTION 256

ACKNOWLEDGEMENTS 258

Figures

FIGURE 1-1: GENERIC SERVER BLADE CONFIGURATION 16
FIGURE 1-2 EXAMPLE SERVER BLADE CONFIGURATION USING MR-IOV TECHNOLOGY 16
FIGURE 1-3: EXAMPLE PLATFORM CONFIGURATION WITHOUT SR-IOV OR MR-IOV TECHNOLOGY 18
FIGURE 1-4: EXAMPLE PLATFORM CONFIGURATION WITH SR-IOV TECHNOLOGY 19
FIGURE 1-5: TWO VIRTUAL HIERARCHIES (VH) IMPLEMENTED OVER SHARED PHYSICAL COMPONENTS 21
FIGURE 1-6: PHYSICAL COMPONENTS THAT CAN BE SUPPORTED IN AN MR-IOV TOPOLOGY 22
FIGURE 1-7: PCIE RP AND MRA PCIE RP FUNCTIONAL BLOCK COMPARISON 23
FIGURE 1-8: PCIE DEVICE, SR-IOV, AND MRA PCIE DEVICE FUNCTIONAL BLOCK COMPARISON 24
FIGURE 1-9: MRA PCIM IN AN MR-IOV TOPOLOGY 27
FIGURE 1-10: PCIE SWITCH AND MRA PCIE SWITCH FUNCTIONAL BLOCK COMPARISON 28
FIGURE 1-11: EXAMPLE MULTI-ROOT TOPOLOGY 30
FIGURE 1-12: EXAMPLE MULTI-ROOT TOPOLOGY AS VIEWED FROM HOST A 32
FIGURE 1-13: EXAMPLE MULTI-ROOT TOPOLOGY AS VIEWED FROM HOST C 33
FIGURE 2-1: MRINIT DLLP FORMAT 34
FIGURE 2-2: EXAMPLE MR TO MR INITIALIZATION SEQUENCE 36
FIGURE 2-3: EXAMPLE MR TO BASE INITIALIZATION 37
FIGURE 2-4: MR DATA LINK CONTROL AND MANAGEMENT STATE MACHINE (MR-DLCMSM) 38
FIGURE 2-5: MR INITFC STATE MACHINE 41
FIGURE 2-6: MRUPDATEFC HEADER DLLP 43

FIGURE 2-7: MRINITFC1_VL HEADER DLLP 43
FIGURE 2-8: MRINITFC1_VH HEADER DLLP 43
FIGURE 2-9: MRINITFC2_VL HEADER DLLP 44
FIGURE 2-10: MRINITFC2_VH HEADER DLLP 44
FIGURE 2-11: MRUPDATEFC DATA DLLP 44
FIGURE 2-12: MRINITFC1_VL DATA DLLP 44
FIGURE 2-13: MRINITFC1_VH DATA DLLP 45
FIGURE 2-14: MRINITFC2_VL DATA DLLP 45
FIGURE 2-15: MRINITFC2_VH DATA DLLP 45
FIGURE 2-16: TLP PREFIX HEADER LOCATION 51
FIGURE 2-17: TLP PREFIX HEADER LAYOUT 51
FIGURE 2-18: MR DATAFLOW EXAMPLES 56
FIGURE 2-19: RESET DLLP EXAMPLE: TOPOLOGY 58
FIGURE 2-20: RESET DLLP EXAMPLE: HOST A'S VIEW 59
FIGURE 2-21: RESET DLLP 61
FIGURE 2-22: UPSTREAM LINK PARTNER RESET SM 64
FIGURE 2-23: DOWNSTREAM LINK PARTNER RESET SM 66
FIGURE 2-24: EXAMPLE MR TOPOLOGY 80
FIGURE 2-25: EXAMPLE MR TOPOLOGY WITH INITIAL LINK DIRECTIONS 82
FIGURE 2-26: EXAMPLE MR TOPOLOGY WITH COMPONENT DISCOVERY DETAILS 85
FIGURE 2-27: EXAMPLE MR TOPOLOGY: RP 0 VIEW 87
FIGURE 2-28: EXAMPLE MR TOPOLOGY: RP 1 VIEW 88
FIGURE 2-29: EXAMPLE MR TOPOLOGY: RP 2 VIEW 88
FIGURE 2-30: EXAMPLE MR TOPOLOGY: RP 3 VIEW 89
FIGURE 2-31: EXAMPLE MR TOPOLOGY: DEVICE L PF/VF MAPPING 93
FIGURE 2-32: EXAMPLE MR TOPOLOGY: DEVICE M VF MAPPING 94
FIGURE 2-33: EXAMPLE MAPPING OF VFS 99
FIGURE 2-34: VF MIGRATION STATE DIAGRAM 101
FIGURE 2-35: INITIAL VF STATE 103
FIGURE 4-1: MR DEVICE CONFIGURATION SPACE 114
FIGURE 4-2: DEVICE MR-IOV CAPABILITY 115
FIGURE 4-3: LVF TABLE 130
FIGURE 4-4: DEVICE FUNCTION TABLE 132
FIGURE 4-5: SWITCH MAPPING TABLES 149
FIGURE 4-6: SWITCH MR-IOV CAPABILITY DIAGRAM 151
FIGURE 4-7: EXAMPLE AUTHORIZATION BITMAP (# VS = 36) 160
FIGURE 4-8: SWITCH PORT TABLE 161
FIGURE 4-9: PORT INTERRUPT STATUS BITMAP 178
FIGURE 4-10: VS TABLE 179
FIGURE 4-11: VS INTERRUPT STATUS BITMAP 183
FIGURE 4-12: VS BRIDGE TABLE 184
FIGURE 4-13: EXAMPLE VL ARBITRATION TABLE WITH 32 PHASES 205
FIGURE 4-14: PERFORMANCE MONITORING AND STATISTICS COLLECTION TABLES 207
FIGURE 6-1: SLOT CAPABILITIES REGISTER 231
FIGURE 6-2: SLOT CONTROL REGISTER 232
FIGURE 6-3: SLOT STATUS REGISTER 234

FIGURE 6-4: PCI EXPRESS CAPABILITIES REGISTER 235
FIGURE 7-1: MULTI-ROOT WAKE-UP SCENARIOS 242
FIGURE 8-1: (VH, VC) TO VL MAPPING 248
FIGURE 8-2: MRA ARBITRATION MODEL 252
FIGURE 8-3: LOGICAL QUEUING STRUCTURE ASSOCIATED WITH A VL RECEIVER 254
FIGURE 8-4: STATISTICS COLLECTION PROCESS 256

Tables

TABLE 2-1: MRINIT DLLP FIELDS 35
TABLE 2-2: MR FLOW CONTROL NEGOTIATION 42
TABLE 2-3: MR FLOW CONTROL DLLP FIELDS 46
TABLE 2-4: RESET DLLP EXAMPLE: EVENTS AND ACTIONS 60
TABLE 2-5: MODIFIED ELECTRICAL IDLE INFERENCE CONDITIONS 72
TABLE 2-6: PORT TYPES – EXAMPLE MR TOPOLOGY 80
TABLE 2-7: EXAMPLE MR TOPOLOGY VH AND VF MAPPING POLICY 86
TABLE 2-8: EXAMPLE TOPOLOGY: SWITCH A VS BRIDGE TABLE CONTENTS 89
TABLE 2-9: EXAMPLE TOPOLOGY: SWITCH B VS BRIDGE TABLE CONTENTS 90
TABLE 2-10: VALID MR STATE TRANSITIONS FOR VF MIGRATION 101
TABLE 4-1: MR-IOV FIELDS 106
TABLE 4-2: DEVICE MR-IOV EXTENDED CAPABILITY HEADER 116
TABLE 4-3: MR-IOV CAPABILITIES 117
TABLE 4-4: DEVICE MR-IOV CONTROL 118
TABLE 4-5: DEVICE MR-IOV STATUS 121
TABLE 4-6: DEVICE MR-IOV VH COUNTS 122
TABLE 4-7: DEVICE FUNCTION TABLE OFFSET 123
TABLE 4-8: VF MVF REGION 124
TABLE 4-9: LVF TABLE OFFSET 124
TABLE 4-10: DEVICE VL ARBITRATION CAPABILITY AND STATUS 125
TABLE 4-11: DEVICE VL ARBITRATION CONTROL 126
TABLE 4-12: DEVICE VL ARBITRATION TABLE OFFSET 127
TABLE 4-13: DEVICE MR ERROR STATUS 128
TABLE 4-14: DEVICE MR ERROR CONTROL 129
TABLE 4-15: LVF TABLE ENTRY 130
TABLE 4-16: FUNCTION CAPABILITY 1 (00H) 133
TABLE 4-17: FUNCTION CAPABILITY 2 (04H) 134
TABLE 4-18: FUNCTION CONTROL 1 (08H) 135
TABLE 4-19: FUNCTION CONTROL 2 (0CH) 136
TABLE 4-20: FUNCTION CONTROL 3 (10H) 138
TABLE 4-21: FUNCTION STATUS 139
TABLE 4-22: FUNCTION TABLE VC TO VL MAP 1 (VC CAPABILITY) 141
TABLE 4-23: FUNCTION TABLE VC TO VL MAP 2 (VC CAPABILITY) 143
TABLE 4-24: FUNCTION TABLE VC RESOURCE STATE 145

TABLE 4-25: VH TABLE MFVC RESOURCE STATE 146
TABLE 4-26: DEVICE PCIE CAPABILITY FIELDS 147
TABLE 4-27: SWITCH MR-IOV EXTENDED CAPABILITY HEADER 152
TABLE 4-28: SWITCH MR-IOV CAPABILITY BITS 152
TABLE 4-29: SWITCH MR-IOV CONTROL BITS 153
TABLE 4-30: SWITCH MR-IOV STATUS BITS 154
TABLE 4-31: SWITCH MR-IOV THIS BRIDGE MAP 154
TABLE 4-32: SWITCH MR-IOV AUTHORIZATION CONTROL 155
TABLE 4-33: SWITCH MR-IOV AUTHORIZATION CONTROL 156
TABLE 4-34: SWITCH PORT TABLE SIZES 156
TABLE 4-35: SWITCH PORT TABLE OFFSET 157
TABLE 4-36: SWITCH VS TABLE SIZES 157
TABLE 4-37: SWITCH VS TABLE OFFSET 158
TABLE 4-38: SWITCH VS BRIDGE TABLE SIZES 158
TABLE 4-39: SWITCH VS BRIDGE TABLE OFFSET 159
TABLE 4-40: SWITCH PORT CAPABILITY 162
TABLE 4-41: SWITCH PORT CONTROL1 163
TABLE 4-42: SWITCH PORT CONTROL2 165
TABLE 4-43: SWITCH PORT STATUS 167
TABLE 4-44: SWITCH LINK PARTNER TRAINING STATUS 168
TABLE 4-45: SWITCH LINK PARTNER TRAINING STATUS – MRINIT DLLP BITS 169
TABLE 4-46: SWITCH VL ARBITRATION CAPABILITY AND STATUS 170
TABLE 4-47: SWITCH VL ARBITRATION CONTROL 171
TABLE 4-48: SWITCH VL ARBITRATION TABLE OFFSET 172
TABLE 4-49: SWITCH MR ERROR STATUS 173
TABLE 4-50: SWITCH MR ERROR CONTROL 174
TABLE 4-51: PCI BRIDGE CONTROL 175
TABLE 4-52: PORT PCIE CAPABILITY STRUCTURE 175
TABLE 4-53: SWITCH VS CAPABILITY AND STATUS 179
TABLE 4-54: SWITCH VS CAPABILITY AND STATUS 181
TABLE 4-55: VS BRIDGE TABLE ENTRIES 185
TABLE 4-56: SWITCH VS BRIDGE CAPABILITY AND STATUS 186
TABLE 4-57: SWITCH VS BRIDGE CONTROL 1 187
TABLE 4-58: SWITCH VS BRIDGE CONTROL 2 190
TABLE 4-59: VIRTUAL HOT-PLUG SIGNALS INTERFACE 1 191
TABLE 4-60: HOT-PLUG SIGNALS INTERFACE 2 192
TABLE 4-61: SWITCH VS BRIDGE VC ID TO VL MAP 1 195
TABLE 4-62: SWITCH VS BRIDGE VC ID TO VL MAP 2 196
TABLE 4-63: VC RESOURCE STATE 198
TABLE 4-64: SWITCH PCIE CAPABILITY FIELDS 199
TABLE 4-65: DEFINITION OF THE 4-BIT ENTRIES IN THE VL ARBITRATION TABLE 205
TABLE 4-66: LENGTH OF THE VL ARBITRATION TABLE 205
TABLE 4-67: STATISTICS TABLE SIZES 208
TABLE 4-68: STATISTICS START/BUSY 209
TABLE 4-69: STATISTICS DESCRIPTOR TABLE OFFSET 210
TABLE 4-70: STATISTICS BLOCK TABLE OFFSET 211

TABLE 4-71: STATISTICS DESCRIPTOR TABLE ENTRY 212
TABLE 4-72: STANDARD STATISTICS 213
TABLE 4-73: TLP FILTERS 216
TABLE 4-74: CREDIT FILTERS 217
TABLE 4-75: DLLP FILTERS 218
TABLE 4-76: STATISTICS BLOCK CAPABILITY 219
TABLE 4-77: STATISTICS TABLE OFFSET 219
TABLE 4-78: STATISTICS WAIT TIME 220
TABLE 4-79: STATISTICS COUNT TIME 220
TABLE 4-80: STATISTICS CAPABILITY AND CONTROL 221
TABLE 4-81: STATISTICS FILTER ENABLE AND CONTROL 222
TABLE 4-82: STATISTICS COUNTER LOW 222
TABLE 4-83: STATISTICS COUNTER HIGH 222
TABLE 5-1: PHYSICAL LAYER ERROR LIST 224
TABLE 5-2: DATA LINK LAYER ERROR LIST 224
TABLE 5-3: TRANSACTION LAYER ERROR LIST 225
TABLE 5-4: MR ERROR LIST 227
TABLE 6-1: VIRTUAL MAPPING: PCIE SLOT CAPABILITIES REGISTER 231
TABLE 6-2: VIRTUAL MAPPING: PCIE SLOT CONTROL REGISTER 233
TABLE 6-3: VIRTUAL MAPPING: PCIE SLOT STATUS REGISTER 234
TABLE 6-4: VIRTUAL MAPPING: PCIE CAPABILITIES REGISTER 235
TABLE 6-5: HOT-PLUG VIRTUAL SIGNALS INTERFACE REGISTER FIELDS 236

Objective of the Specification

The purpose of this document is to specify PCI® I/O virtualization and sharing technology. The specification is focused on multi-root topologies; e.g., a server blade enclosure that uses a PCI Express® Switch-based topology to connect server blades to PCI Express Devices or PCI Express-to-PCI Bridges and enable the leaf Devices to be serially or simultaneously shared by one or more server blades.

This document is to be used in conjunction with, and does not supersede, the terms and conditions specified in the PCI-SIG® Trademark and Logo Usage Guidelines document.

Document Organization

Chapter 1 specifies the architectural overview defining the basic building blocks of the technology.

Chapter 2 specifies the multi-root protocol changes that build upon the *PCI Express Base Specification.*

Chapter 3 specifies multi-root initialization algorithms and requirements.

Chapter 4 specifies multi-root configuration and control structures.

Chapter 5 specifies multi-root error handling and coordination across multiple virtual hierarchies.

Chapter 6 specifies multi-root hot-plug management controls

Chapter 7 specifies multi-root power management and coordination across multiple virtual hierarchies.

Chapter 8 specifies multi-root congestion management services and requirements.

Documentation Conventions

Capitalization

Some terms are capitalized to distinguish their definition in the context of this document from their common English meaning. Words not capitalized have their common English meaning. When terms such as "memory write" or "memory read" appear completely in lower case, they include all transactions of that type.

Register names and the names of fields and bits in registers and headers are presented with the first letter capitalized and the remainder in lower case.

Numbers and Number Bases

Hexadecimal numbers are written with a lower case "h" suffix, e.g., 0FFFFh and 80h. Hexadecimal numbers larger than four digits are represented with a space dividing each group of four digits, as in 1E FFFF FFFFh. Binary numbers are written with a lower case "b" suffix, e.g., 1001b and 10b. Binary numbers larger than four digits are written with a space dividing each group of four digits, as in 1000 0101 0010b.

All other numbers are decimal.

Reference Information

Reference information is provided in various places to assist the reader and does not represent a requirement of this document. Such references are indicated by the abbreviation "(ref)." For example, in some places, a clock that is specified to have a minimum period of 400 ps also includes the reference information maximum clock frequency of "2.5 GHz (ref)." Requirements of other specifications also appear in various places throughout this document and are marked as reference information. Every effort has been made to guarantee that this information accurately reflects the referenced document; however, in case of a discrepancy, the original document takes precedence.

Implementation Notes

Implementation Notes should not be considered to be part of this specification. They are included for clarification and illustration only. Implementation Notes within this document are enclosed in a box and set apart from other text.

Terms and Abbreviations

Base Function (BF)	Function used to manage the MR features of an MR Device.
Single Root I/O Virtualization (SR-IOV)	The capability for a single PCIe® component to be used by more than one SI. This functionality is defined in the *Single Root I/O Virtualization and Sharing Specification*.
Switch Management Port	A connection to an MR Switch that can be used to manage an MR topology. Switch Management Ports may be actual PCIe Ports or may be Vendor Specific non-PCIe Ports.
Multi Root I/O Virtualization (MR-IOV)	The capability for a single PCIe® component to be used by more than one Hierarchy Domain. Additionally, for Root Ports, the capability for a single Root Port to support more than one Hierarchy Domain.
MR - Multi-Root Topology	A PCIe topology that interconnects one or more Root Ports through multi-root aware Switches.
MR PCIM	Multi-Root PCIM.
MRA	Multi-Root Aware. A PCIe component that supports the multi-root extensions defined in this specification.
MRA VF	Virtual Function (VF) in an MRA Device.
MR Enabled Link	PCIe Link using the Multi-Root Encapsulated Link Protocol.
MR Egress	Point in a PCIe fabric that a TLP exits the MR Fabric.
MR Fabric	Subset of a PCIe fabric containing MR Enabled Links and connected MRA components.
MR Ingress	Point in a PCIe fabric that a TLP enters the MR Fabric.
PCI Bus Memory Address	Refers to the address portion of a PCI Memory Transaction.
PCI Manager (PCIM)	Software that enumerates and configures an IOV topology.
PCIe	PCI Express.
Physical Address	The address used by the system memory controller to access system memory.

Physical Function (PF)	An IOV-capable Function per the *Single Root I/O Virtualization and Sharing Specification*. In MR, PFs exist within Virtual Hierarchies (VHs).
RC	Root Complex per the *PCI Express Base Specification.*
RP	Root Port per the *PCI Express Base Specification.*
System Image (SI)	A software component running on a Virtual System to which specific virtual and physical Devices can be assigned. Specification of the behavior and architecture of an SI is outside the scope of this specification. Examples of SIs include guest operating systems and shared/non-shared protected domain Device drivers.
SR PCIM	Single Root PCIM.
Virtual Device	Collection of PFs and VFs that operate as a Device within a VH.
Virtual Function (VF)	An IOV-capable Function per the *Single Root I/O Virtualization and Sharing Specification*. In MR, VFs exist within Virtual Hierarchies (VHs).
Virtual Hierarchy (VH)	Portion of an MR Topology assigned to a single PCIe Domain Hierarchy.
Virtual Hierarchy Number (VHN)	Link Local Number designating a VH.
Virtual Intermediary (VI)	A software component supporting one or more SIs – colloquially known as a Hypervisor or Virtual Machine Monitor. Specification of the behavior and architecture of VI is outside the scope of this specification.
Virtual Switch (VS)	A logical PCIe Switch associated with a single VH implemented in an MR Switch.

1

1. Architectural Overview

Within the industry, significant effort has been expended to increase the effective hardware resource utilization through the use of virtualization technology. The Multi-Root I/O Virtualization (MR-IOV) specification defines the extensions to the PCI Express (PCIe) specification suite to enable multiple non-coherent Root Complexes (RCs) to share PCI hardware resources.

To illustrate how this technology can be used to increase effective resource utilization, consider the following generic server blade configuration as illustrated in the Figure 1-1 below.

- The server blade configuration contains four server blades and two external fabric switches. In a high availability configuration, nominally there would be two external fabric switches of each type to avoid any single point of failure (SPOF) for a total of four switches.
- In this example, each switch provides two external connectivity Ports though more can be configured to deliver high availability solutions as well as increased aggregate performance.
- Each server blade is provisioned with two PCIe Endpoint Devices – an Ethernet and a storage area network (SAN) device. This translates to a total of eight PCIe Endpoint Devices. These PCIe Devices are point-to-point connected to a Root Port (RP) [not shown] – either emitted by a chipset or a processor.
- Each I/O device and Switch Port is typically provisioned to enable any I/O device to operate at full bandwidth.

Depending upon workload, the example configuration's I/O resource capacity may be excessive resulting in under-utilized hardware.

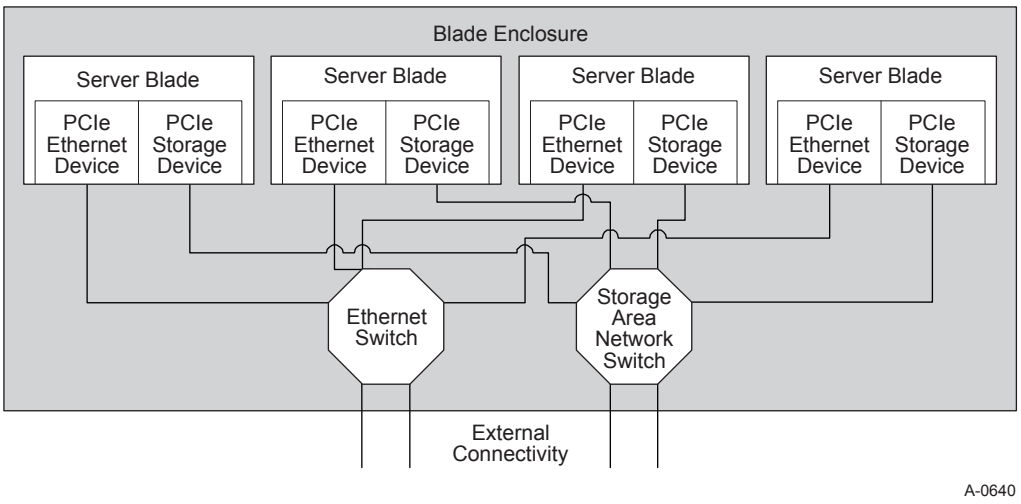

Figure 1-1: Generic Server Blade Configuration

Through the application of MR-IOV technology, the prior example server blade configuration can be transformed as illustrated in Figure 1-2.

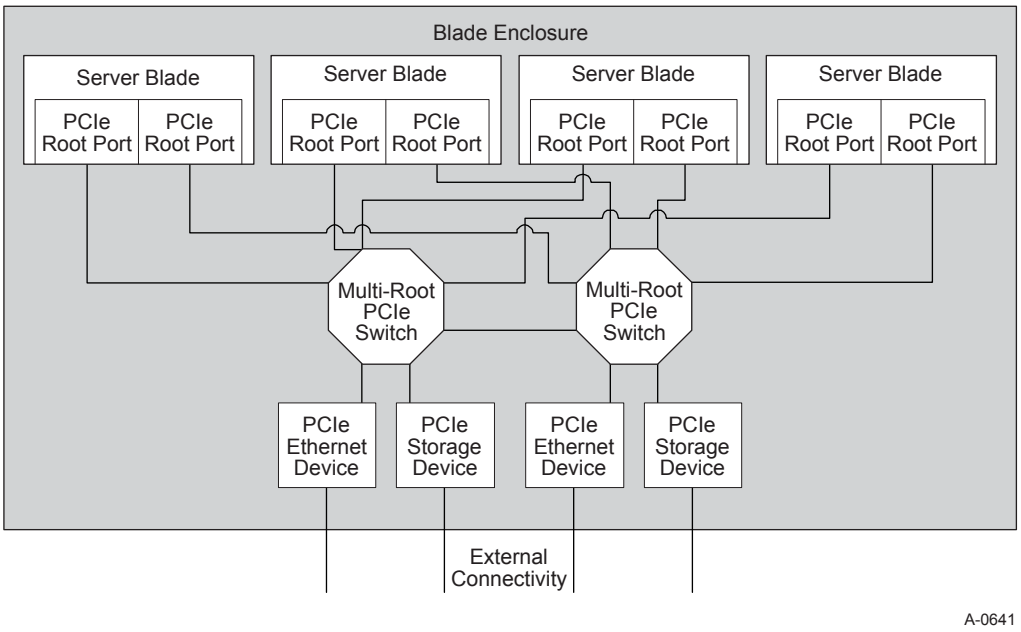

Figure 1-2 Example Server Blade Configuration Using MR-IOV Technology

In contrast to the Figure 1-1, the following can be observed:

- The server blade configuration contains four server blades. The server blades do not contain PCIe Endpoint Devices but instead connect a Root Port (RP) to a Multi-Root Aware (MRA) PCIe Switch.
- The two external fabric switches are replaced by two MRA PCIe Switches. While the details will be described in a subsequent section, the following should be noted:

- Unlike a PCIe Switch which contains a single upstream Port and can only be claimed by a single RP, an MRA PCIe Switch contains multiple upstream Ports to enable it to connect to multiple RPs. This enables the MRA PCIe Switch to be a shared component within the configuration.
- Multiple MRA PCIe Switches can be interconnected in a variety of topologies to create high availability solutions as well as provide increased I/O fan-out capacity.

- In place of eight PCIe Endpoint Devices – four of each type – the example MR-IOV configuration contains four MRA PCIe Endpoint Devices – two of each type. Each MRA PCIe Endpoint Device is attached to an MRA PCIe Switch downstream Port enabling each to be accessed, and thus shared, by any of the server blades.
- Unlike the prior example configuration where I/O is dedicated to each server blade, an MR-IOV based configuration enables the I/O to be dynamically assigned. A fraction or an entire I/O Device can be assigned to each server blade based on its workload requirements.

As noted above, an MR-IOV configuration reduces the component count and changes the component composition. This specification covers the elements involved in delivering an MR-IOV configuration.

1.1. How Does MR-IOV Work?

To understand how MR-IOV works, first examine an example platform configuration devoid of any Single Root IOV (SR-IOV)–see the *Single Root I/O Virtualization and Sharing Specification*–or MR-IOV technology as illustrated in Figure 1-3.

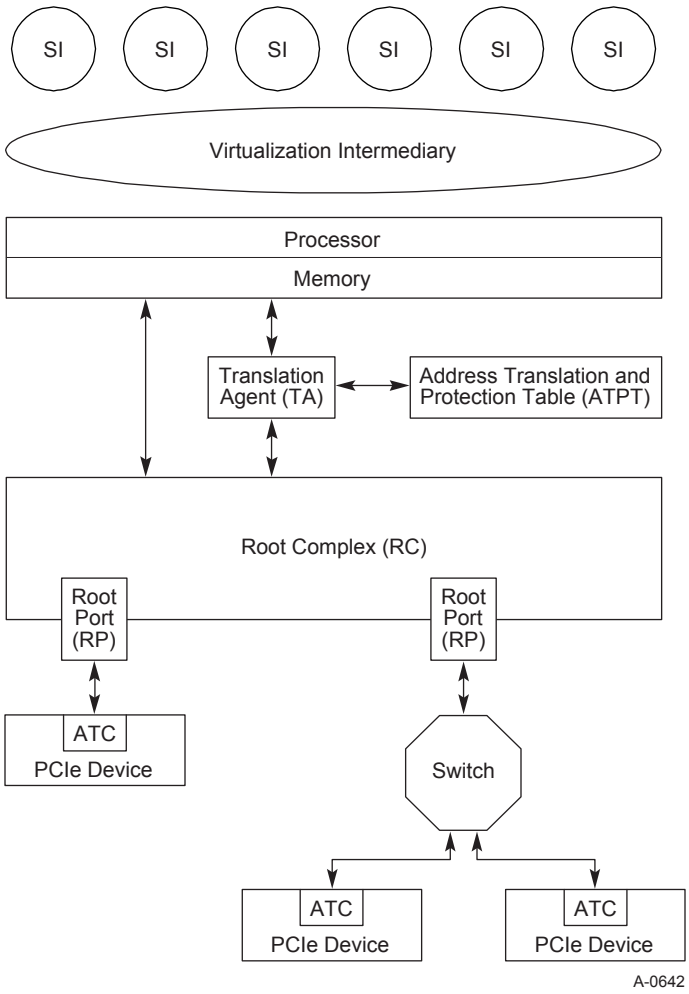

Figure 1-3: Example Platform Configuration Without SR-IOV or MR-IOV Technology

The above example platform is composed of the following:

- A processor capable of running any operating system. In virtualized environments, a processor can execute a Virtual Intermediary (VI) which abstracts or virtualizes all hardware from one or more System Images (SI). Each SI contains a virtual I/O device driver while the VI will contain the I/O-specific device drivers and perform all I/O hardware accesses.
- A memory controller and associated memory.
- A Translation Agent (TA) and Address Translation and Protection Table (ATPT).
  - A TA parses the contents of a PCIe DMA request transaction (TLP) to index an ATPT to derive the physical address translation and access rights. The purposes for having DMA address translation vary and include:
    - Limiting the destructiveness of a "broken" or miss-programmed DMA I/O Function
    - Providing for scatter/gather
    - Ability to redirect message-signaled interrupts (e.g., MSI or MSI-X) to different address ranges without requiring coordination with the underlying I/O Function

- ♦ Address space conversion (32-bit I/O Function to larger system address space)
- ♦ Virtualization support

- A PCIe Endpoint may contain an Address Translation Cache (ATC) in support of the PCI-SIG *Address Translation Services Specification*.

❑ A PCIe Root Complex (RC) containing one or more Root Ports (RP) with direct-attached or PCIe Switch-attached PCIe Devices or PCI/PCI-X Bridges. Each RP defines a unique hierarchy domain (see the *PCI Express Base Specification*).

Now examine a platform that supports SR-IOV technology as illustrated in Figure 1-4 below.

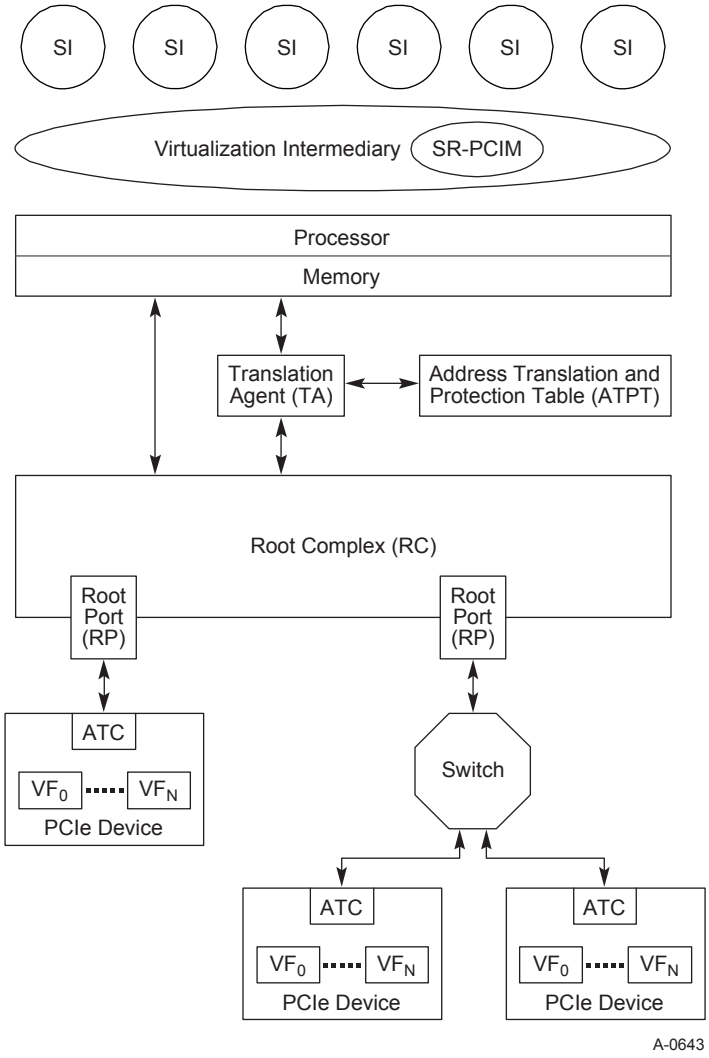

Figure 1-4: Example Platform Configuration with SR-IOV Technology

The differences between the platforms in Figure 1-3 and Figure 1-4 are:

❑ The PCIe Devices support the SR-IOV capability as defined in the *Single Root I/O Virtualization and Sharing Specification*. SR-IOV enables a PCIe Device to support multiple Virtual Functions (VFs).

- SR-PCIM (Single Root PCI Manager) is responsible for SR-IOV hardware configuration and management. SR-PCIM is nominally integrated within a VI, though a variety of implementation options exist and remain outside the scope of the PCI-SIG specifications.

From the prior two figures, the following key semantics should be kept in mind:

1. The operating system or VI has exclusive control over each PCIe Component – RC, RP, Switch, Link, PCIe-to-PCI/PCI-X Bridge, Device, and Function.
    a. All PCI enumeration, configuration operations, reset, power management, and event handling, e.g. errors, must only be initiated or processed by the operating system or VI. Operations initiated by a SI are trapped by the VI and processed on its behalf.
2. Each RP acts as the terminus for all upstream targeted operations, e.g., error event notification.

In order to support either example platform while preserving these semantics, the PCI components underneath each RP must be virtualized and logically overlaid on the MRA PCIe Switches and Devices as illustrated in Figure 1-5. The virtualized PCI components are referred to as a Virtual Hierarchy (VH). A VH has the following attributes:

- Each VH must contain at least one PCIe Switch.
    - The PCIe Switch will be a virtualized component implemented using a Virtual Switch within an MRA Switch.
    - The PCIe Switch functionality and semantics are per the *PCI Express Base Specification*.
- Each VH may contain any mix of PCIe Devices, MRA PCIe Devices, or PCIe to PCI/PCI-X Bridges as illustrated in Figure 1-6.
    - A PCIe Device is a device that does not support the MR-IOV Capability. Such a device must only be visible in a single VH at a time. A PCIe Device can be serially shared among a set of accessible VH within the MR-IOV topology. The PCIe Device is serially deleted from the current source VH and added to destination VH.
    - A PCIe Device may support the SR-IOV Capability which enables it to be shared by multiple SI executing above a single RP.
    - A PCIe to PCI/PCI-X Bridge can only be visible in a single VH at a time. As with a PCIe Device, a PCIe to PCI/PCI-X Bridge can be serially shared among a set of accessible VH within the MR-IOV topology using a conceptually similar deletion/addition process as a PCIe Device.
        - The SR-IOV Capability does not apply to the PCIe to PCI/PCI-X Bridge. The bridge and all associated PCI/PCI-X devices can only be configured in a single operating system, VI, or SI at a time.
    - An MRA PCIe Device is a device which supports the MR-IOV Capability. Such a device can be visible in multiple VHs at a time depending upon the MR-IOV resources provisioned. An MRA PCIe Device can be added or deleted from any accessible VH within an MR-IOV topology.
    - An MRA PCIe Device may support the SR-IOV Capability in each VH. This enables it to be shared by multiple SI executing above each RP.

- The MR-IOV topology typically contains at least one MRA PCIe Switch.[1]
  - Multiple MRA PCIe Switches can be provisioned and interconnected in a variety of topologies – tree, fat-tree, star, mesh, etc.
  - Virtual Switches within an MRA PCIe Switch must contain a virtual upstream Port. This virtual upstream Port must connect, directly or indirectly, to a RP which acts as the root of the VH.

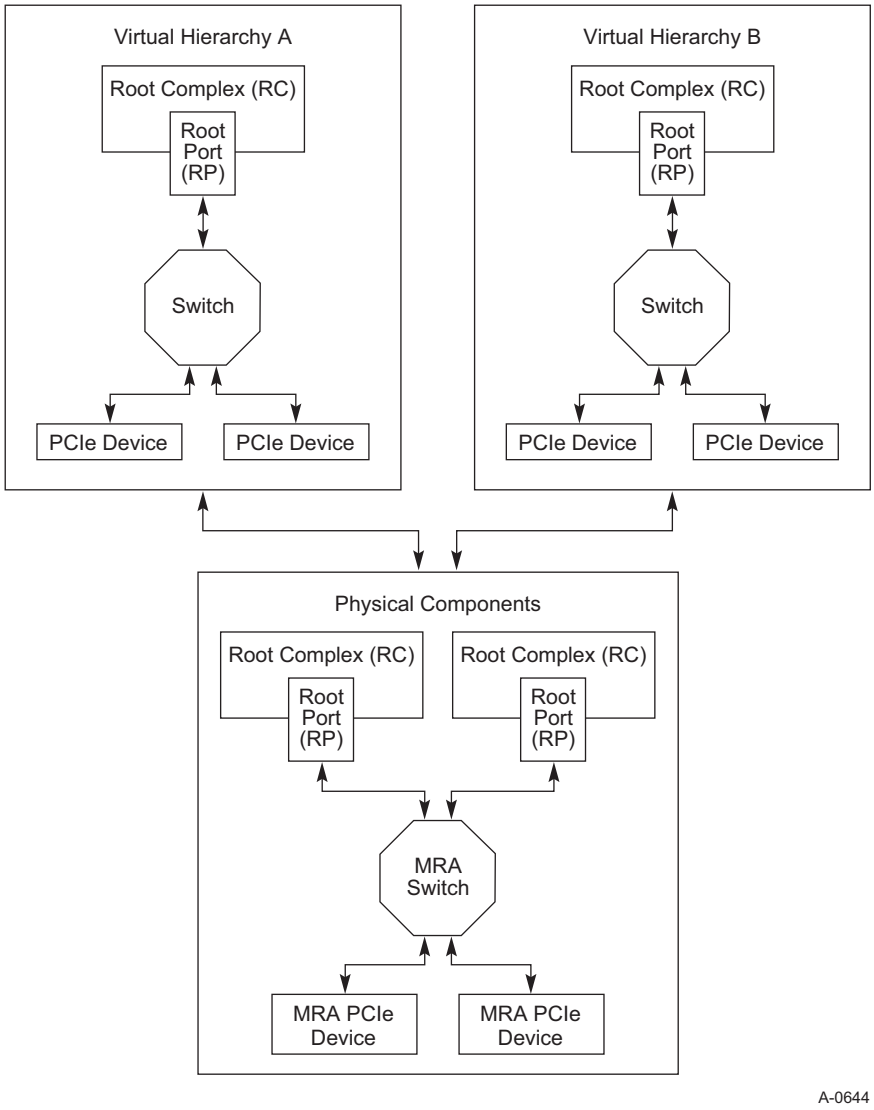

Figure 1-5: Two Virtual Hierarchies (VH) Implemented Over Shared Physical Components

[1] Topologies incorporating MRA Root Complexes need not include an MRA Switch.

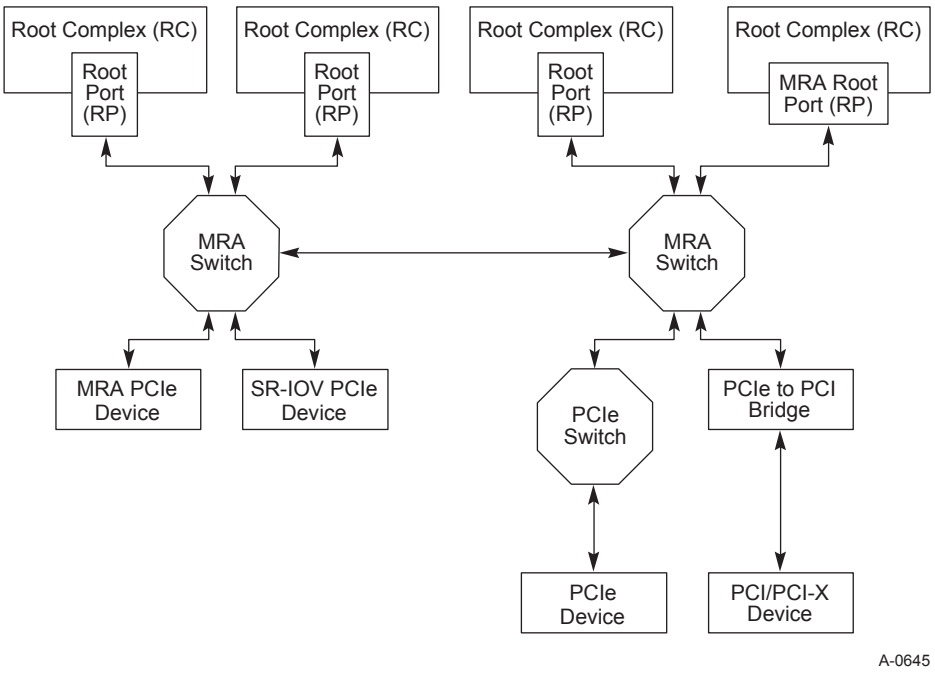

Figure 1-6: Physical Components that can be Supported in an MR-IOV Topology

1.1.1. MRA Components

The prior section illustrated that MR-IOV is primarily the overlaying of multiple VH over a shared physical set of MRA and non-MRA components. To further understand how MR-IOV works, we will examine the MRA components in more detail.

1.1.1.1. Multi-Root Aware Root Port (MRA RP)

As illustrated in Figure 1-6, a PCIe RP supporting either a single operating system or a VI with multiple SI can be connected to an MRA Switch and access multiple downstream devices and bridges. The PCIe RP, though, is restricted to a single VH. In order to enable multiple VH to be accessed, an MRA PCIe RP is required. An MRA PCIe RP differs from a PCIe RP in the following ways:

- An MRA PCIe RP maintains state to delineate each VH. At a high level, this amounts to a set of resource mapping tables to translate the I/O function associated with each SI into a VH and MR I/O function identifier.
- An MRA PCIe RP participates in the MR transaction encapsulation protocol (see Section 1.1.1.2 for details) to enable an MRA PCIe Switch to derive the VH and associated routing information.
- An MRA PCIe RP emits an MRA Link. An MRA Link is identical to the physical layer of a PCIe Link as defined in the *PCI Express Base Specification*. An MRA Link differs at the Data Link Layer where a new set of DLLPs are defined to support the MR-IOV protocol.

- An MRA PCIe RP may implement MRA congestion management (see Section 2.4 and Chapter 8 for details).
- An MRA Root Port does not forward TLPs for VHs where it is not the root. If this functionalityis required, the MRA Root Complex may contain an embedded MRA Switch.

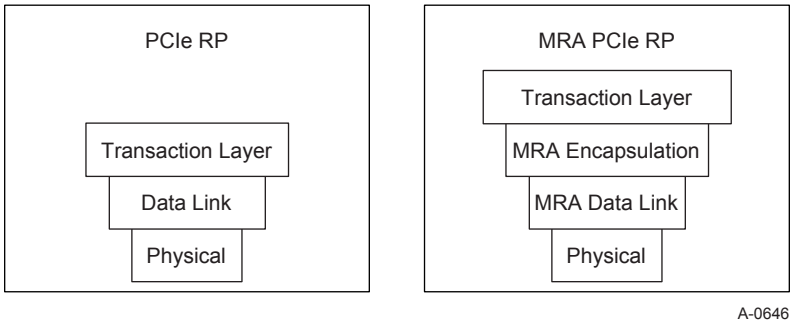

Figure 1-7: PCIe RP and MRA PCIe RP Functional Block Comparison

1.1.1.2. Multi-Root Aware PCIe Device (MRA PCIe Device)

An MR-IOV platform may contain any mix of PCIe Devices, SR-IOV PCIe Devices, or MR-IOV PCIe Devices. Figure 1-8 illustrates a functional block comparison between these three types of devices.

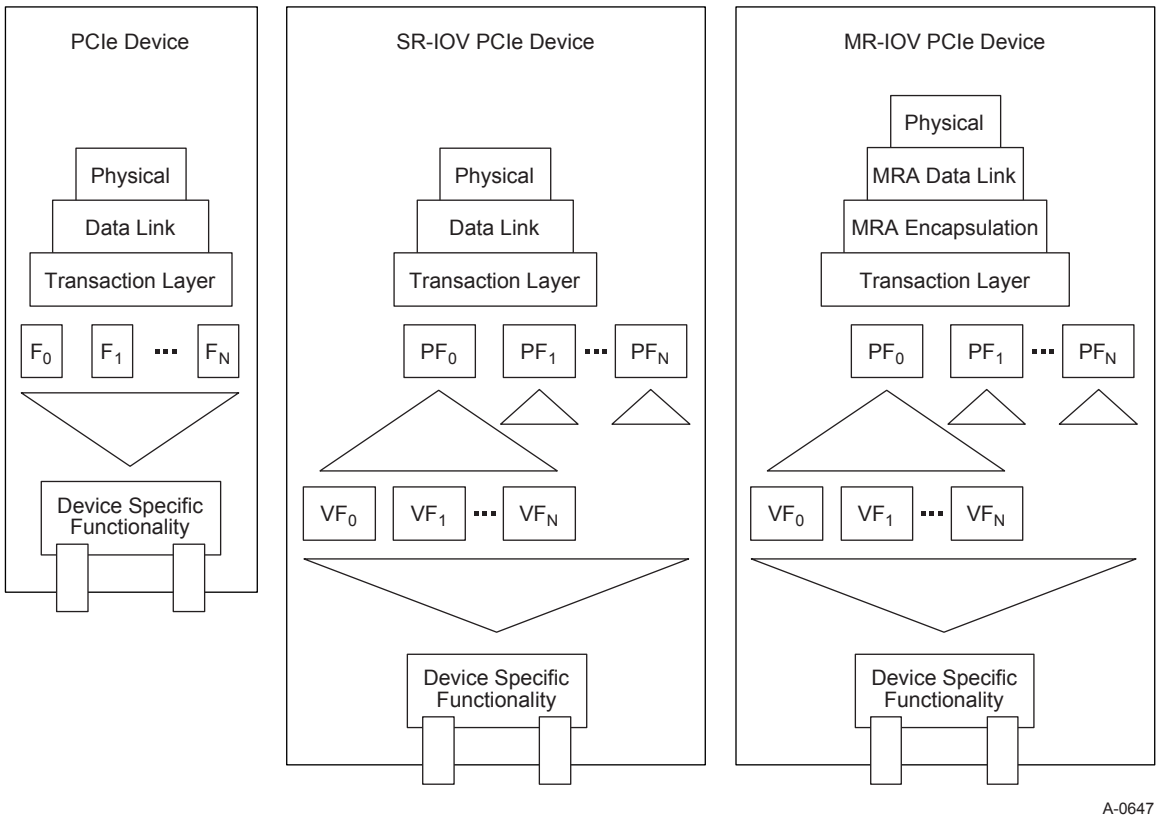

Figure 1-8: PCIe Device, SR-IOV, and MRA PCIe Device Functional Block Comparison

An MRA IOV PCIe Device differs from a PCIe Device and a SR-IOV PCIe Device in the following ways:

- The MRA IOV PCIe Device must support the new MR-IOV DLLP protocol.
  - A PCIe Device or an SR-IOV PCIe Device do not support the MR-IOV capability and therefore are unable to participate in this protocol. An MRA PCIe Switch must subsume all responsibility for forwarding transactions and event handling on behalf of these devices through the MR-IOV topology. The MRA PCIe Switch will perform all encapsulation or de-encapsulation as appropriate.
- The MRA IOV PCIe Device must support the MR-IOV transaction encapsulation protocol.
  - The MR-IOV encapsulation protocol provides VH identification information to the MRA PCIe Switch to enable the transaction to be transparently forwarded through the MR-IOV topology without requiring modification to the *PCI Express Base Specification* TLP protocol or contents.
- The MRA IOV PCIe Device is composed of a set of Functions in each VH.
- There are a variety of Function types:
  - A BF is a Function compliant with this speciation that includes the MR-IOV Capability. A BF may not contain an SR-IOV Capability.
    - The BF is used by MR-PCIM to manage the sharing of the device.

- ♦ The BF may also be used to mange sharing the device specific portions of the device (e.g. MAC addresses, VLAN tags, etc.). If this is done, a tight coupling between MR-PCIM and device specific software may be required.
- ♦ Resetting a BF resets all functions (PFs and Plain Functions) that are associated with that BF.

- A PF is a Function compliant with the *PCI Express Base Specification* that includes the SR-IOV Extended Capability. Every PF is associated with a BF. The Function Offset fields in a BF's Function Table point to the PFs.
- A VF is a Function associated with a PF and is described in the *Single-Root I/O Virtualization and Sharing Specification*. VFs are associated with a PF and are thus indirectly associated with a BF.
- A "plain" function is a Function that is not a BF, PF or VF. Plain Functions may or may not be associated with a BF.

- ❑ VH0 of an MRA IOV PCIe Device contains:
  - One or more Base Functions (BFs) ***bf 0:f***.[2]
  - One optional Physical Function (PF) ***pf 0:f*** or Plain Function ***fcn 0:f*** associated with each BF.
  - Zero or more VFs ***vf 0:f,n*** associated with each PF.[3]
  - Zero more Plain Functions ***fcn 0:f*** that are not associated with a BF.
- ❑ Every Non-zero VH ***h*** of an MRA IOV PCIe Device contains:
  - One Physical Function (PF) ***pf h:f*** or Plain Function ***fcn h:f*** associated with each BF.
  - Zero more VFs ***vf h:f,n*** associated with each PF.
- ❑ The actual function numbers used for BFs, PFs, VFs and Plain Functions are not defined by this specification. The usual PCIe rules apply (i.e. every Device must have a Function 0 in every VH; ARI support is mandatory so function numbers for PFs, BFs and Functions must be between [0..255]; function numbers for VFs can be between [0..255] and can spill onto additional bus numbers).
- ❑ The number of VFs provisioned per PF may vary on a per PF basis.
- ❑ Each PF represents a single device-specific functionality; e.g., an Ethernet controller, a SATA controller, etc. Subsequently, each VF must represent the same device-specific functionality. This enables the existing device driver models to be supported.

---

[2] Nomenclature used for BFs, PFs and Plain Functions is ***type h:f*** where ***type*** is either ***bf***, ***pf*** or ***fcn***, where ***h*** represents the VH number and where ***f*** represents the function number on the captured bus number.

[3] Nomenclature used for VFs is ***vf h:f,n*** for the ***n***th VF associated with ***pf h:f***.

1.1.1.3. Multi-Root PCI Manager (MR-PCIM)

Each MRA component must support a corresponding MR-IOV capability. This capability is accessed and configured by the Multi Root PCI Manager (MR-PCIM). MR-PCIM can be implemented anywhere within the MR-IOV topology; for example, above a RP as illustrated in Figure 1-9, or, for example, through a private interface provided by an MRA PCIe Switch.

Responsibilities of MR-PCIM include:

- Enumeration of the physical components within the MR-IOV topology. MR-PCIM must determine what components are or are not MR-IOV capable, how components are interconnected, and what PCIe and MR-IOV resources they provide.
- MR-PCIM configures the components and resources that comprise each VH. The policies to determine this are outside the scope of this specification.
- Given the physical hardware is shared among a set of VH, MR-PCIM configures all PCIe and MR-IOV attributes including: Link signaling rate, VC arbitration, etc.
- Given the physical hardware is shared among a set of VH, MR-PCIM processes or controls various events; e.g., RESET, physical hardware failure, surprise add/remove, error handling, etc.
- PCI Express Hot Plug can be used to gracefully remove and add virtual Devices and virtual Switches to VHs. MR-PCIM controls this process.
- MR related errors are recorded and made available to MR-PCIM.
- Various statistics gathering facilities are provided for MR-PCIM to monitor performance.

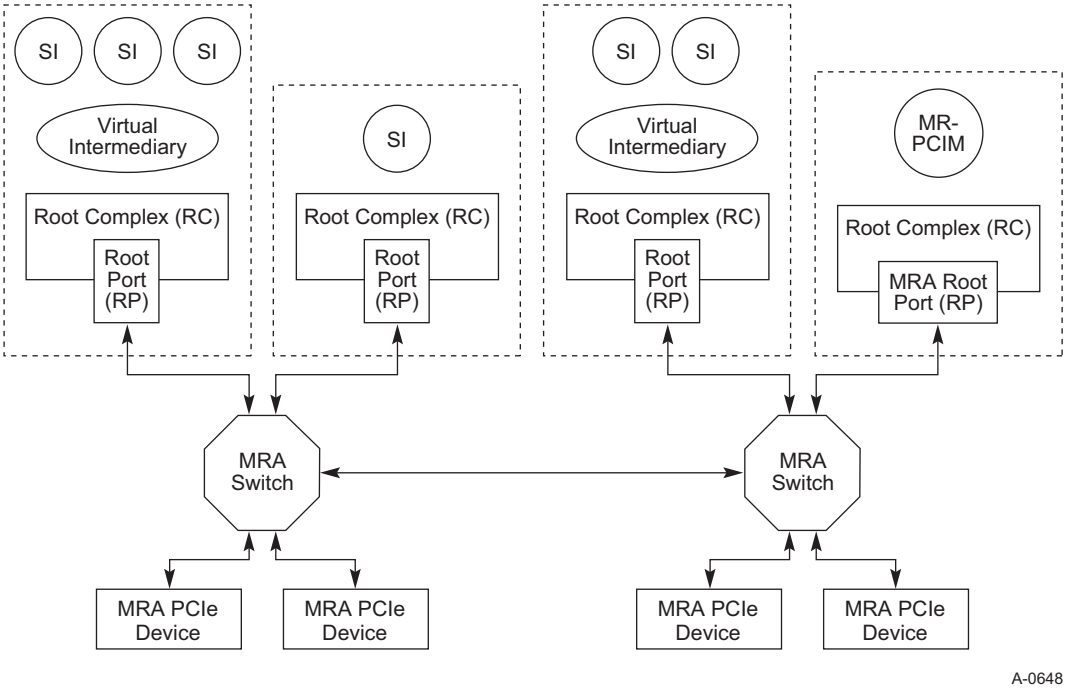

Figure 1-9: MRA PCIM in an MR-IOV Topology

1.1.1.4. Multi-Root Aware PCIe Switch (MRA PCIe Switch)

In a prior section, it was noted that an MRA Switch is conceptually the overlay of multiple PCIe Switches onto a single physical package. This is illustrated in more detail in Figure 1-10.

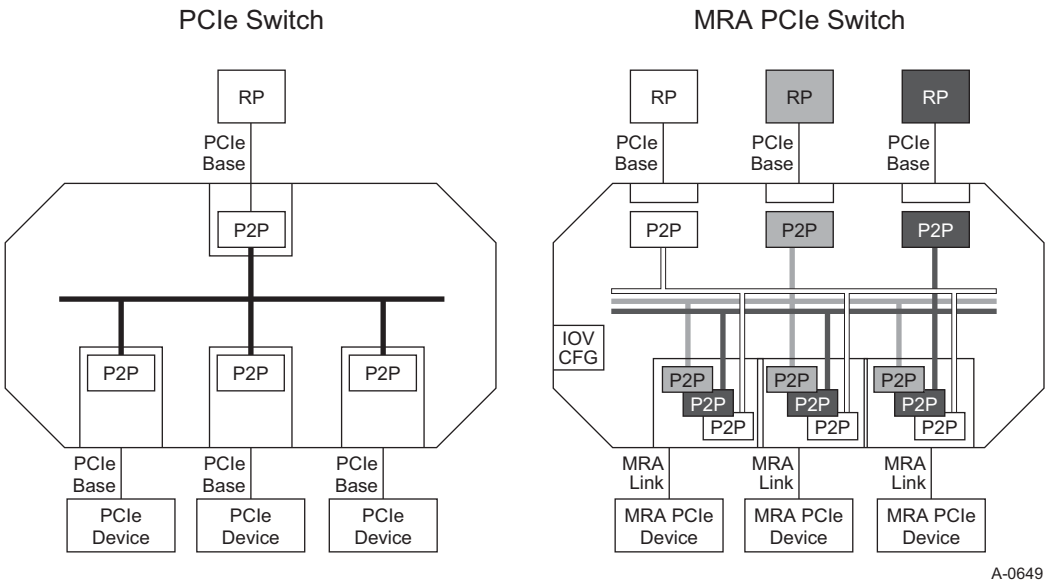

Figure 1-10: PCIe Switch and MRA PCIe Switch Functional Block Comparison

The PCIe Switch is composed of a set of logical P2P bridges with a single upstream Port attached to a PCIe RP and one or more downstream Ports attached to either a PCIe Device or a PCIe to PCI/PCI-X Bridge. A PCIe Switch also operates using a single address space.

In contrast to a PCIe Switch, an MRA Switch is as follows:

- An MRA Switch is composed of one or more upstream Ports attached to either a PCIe RP or an MRA PCIe RP or the downstream Port of an MRA Switch.
  - If the upstream Port is attached to a PCIe RP, the MRA Switch must transparently provide all MRA related services on behalf of the PCIe RP.
- An MRA Switch is composed of one or more downstream Ports attached to PCIe Devices, MRA PCIe Devices, PCIe Switch upstream Ports, MRA Switch upstream Ports, or PCIe to PCI/PCI-X Bridges.
- A set of logical P2P bridges that constitute a VH.
- Each VH represents a separate address space. The combination of a VH identifier and the address contained within the PCIe TLP enable the MRA Switch to forward the TLP to an appropriate egress Port as well as an MRA RP or MRA PCIe Device to delineate which PF or VF is the source or sink of the PCIe TLP.

1.1.2. MR-IOV and ARI (Alternative Routing Identifier)

Alternative Routing Identifier (ARI) is a feature that allows a PCI Express component to support up to 256 Functions. Without ARI, the component is limited to 8 Functions.

MR Switches must implement ARI Forwarding in all Downstream VS Bridges and may implement the ARI Capability in Upstream VS Bridges.

MR Devices must implement the ARI Capability. ARI support is also required by the *Single Root I/O Virtualization and Sharing Specification*.

Note: Presence of the ARI Capability implies certain behavior in addition to allowing more than 8 Functions.

ARI support is not required for non-MR components used in an MR Topology.

1.1.3. MR-IOV Relationship to SR-IOV and ATS

Briefly:

- An MRA PCIe Device must support the SR-IOV Capability per the *Single Root I/O Virtualization and Sharing Specification*.
  - A SR-IOV PCIe Device must support the *PCI Express Base Specification*.
  - By requiring these specifications to be supported, the number of permutations is reduced further enhancing the ability to deploy and interoperate across a wide range of solution options.
- An MRA PCIe Device may support the *Address Translation Services Specification*.

1.2. Overview of MR Transaction Layer

Figure 1-11 shows an example Multi-Root Topology.

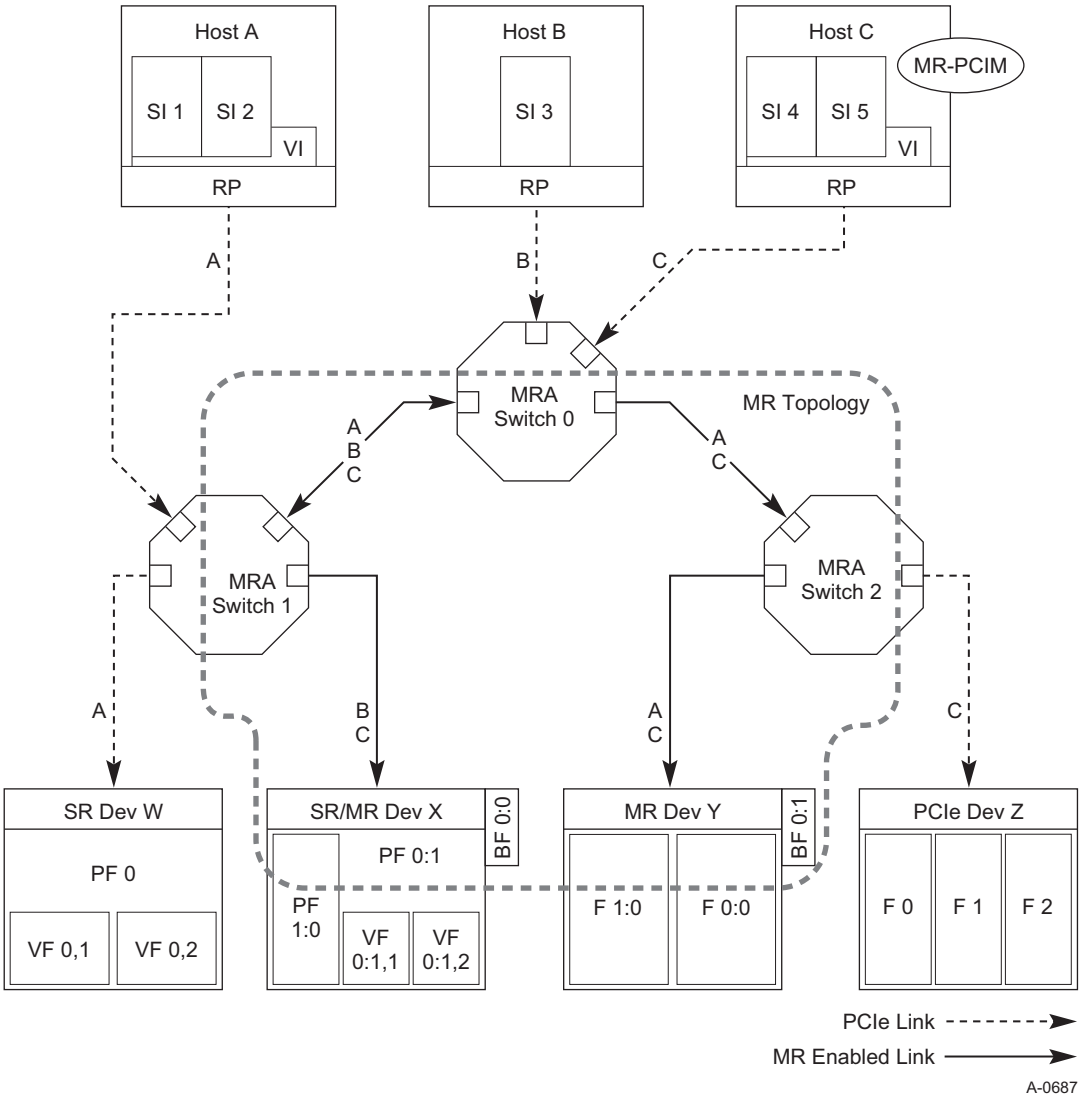

Figure 1-11: Example Multi-Root Topology

The solid links use the Multi-Root Wire Protocol described in this specification. The dashed links use the PCIe Protocol. Functions on SR Devices are designated **PF *f*** and **VF *f,s*** where ***f*** represents the Function Number of the PF and ***s*** indicates which VF Slot belonging to the PF is involved. Functions on MR Devices are designated **PF *h:f,*** **VF *h:f,s*** or **F *h:f*** where ***h*** is added to indicate which Virtual Hierarchy (VH) is involved.

This example shows a single Multi-Root Topology. TLPs inside the MR Topology are labeled with the VH they belong to. TLPs outside the MR Topology belong to a single VH and no label is needed. The MR Ingress point is the point where a TLP first encounters an MR Topology. The

MR Egress point is where a TLP exits an MR Topology. These points exist inside some MR component (in this example, the Switches and Devices X and Y).

In this example, all Root Ports use PCIe protocol. Each Root Port is the root of a PCIe Hierarchy. There are three VHs (A, B, C) each associated with one of the root Ports. Hosts A and C are running a Virtual Intermediary that supports SR sharing. B is running an operating system directly (no Virtual Intermediary is involved).

In this example, there are four Devices, one of each variety.

- Device W is a Single-Root IOV Device. It is assigned to VH A. Two System Images (SIs) are in use on Host A. The Virtual Intermediary on Host A has further assigned VF 0,1 to SI 1 and VF 0,2 to SI 2.
- Device X is using both Single Root and Multi-Root sharing. PF 1:0 is assigned to VH B. PF 0:1 is assigned to VH C. The Virtual Intermediary running on Host C has further assigned VF 0:1,1 to SI 4 and VF 0:1,2 to SI 5. The MR features of the device are managed through the Base Function which is assigned to VH C.
- Device Y is using only Multi-Root sharing. F 1:0 is assigned to VH A and F 0:0 is assigned to VH C. In VH A, the Virtual Intermediary has further assigned F 1:0 to SI 2. In VH C, the Virtual Intermediary has further assigned F 0:0 to SI 4. The MR features of the device are managed through the Base Function which is assigned to VH C.
- Device Z is a 3 Function PCIe Device. It is assigned to VH C. Virtual Intermediary software on VH C has further assigned F 0 and F 1 to SI 4 and F 2 to SI 5.

All Switches shown in this example are Multi-Root Aware (MRA). Non-MRA Switches are also possible; however, such Switches and all components below them will be associated with a single Root Port in a single VH. Note that this non-MR sub-tree can be a mixture of SR aware and non-SR aware PCIe components.

Multi-Root Aware Components enforce separation between VHs. Software running in one VH is not allowed to affect other VHs. For example, every VH has a complete and independent address space.

Figure 1-12 shows the same example as Figure 1-11 but only shows the components visible to Host A. The MRA Switches and Devices appear to software as Single Root equivalents.

Similarly, Figure 1-13 also shows the example from Figure 1-11 but shows only components visible to Host C. Note that the Link between Switch 0 and Switch 1 changes direction between these views of the topology. In Multi-Root systems, the logical upstream/downstream direction of a Link is a per-VH concept and is distinct from physical Link direction that was established during Link bring up.

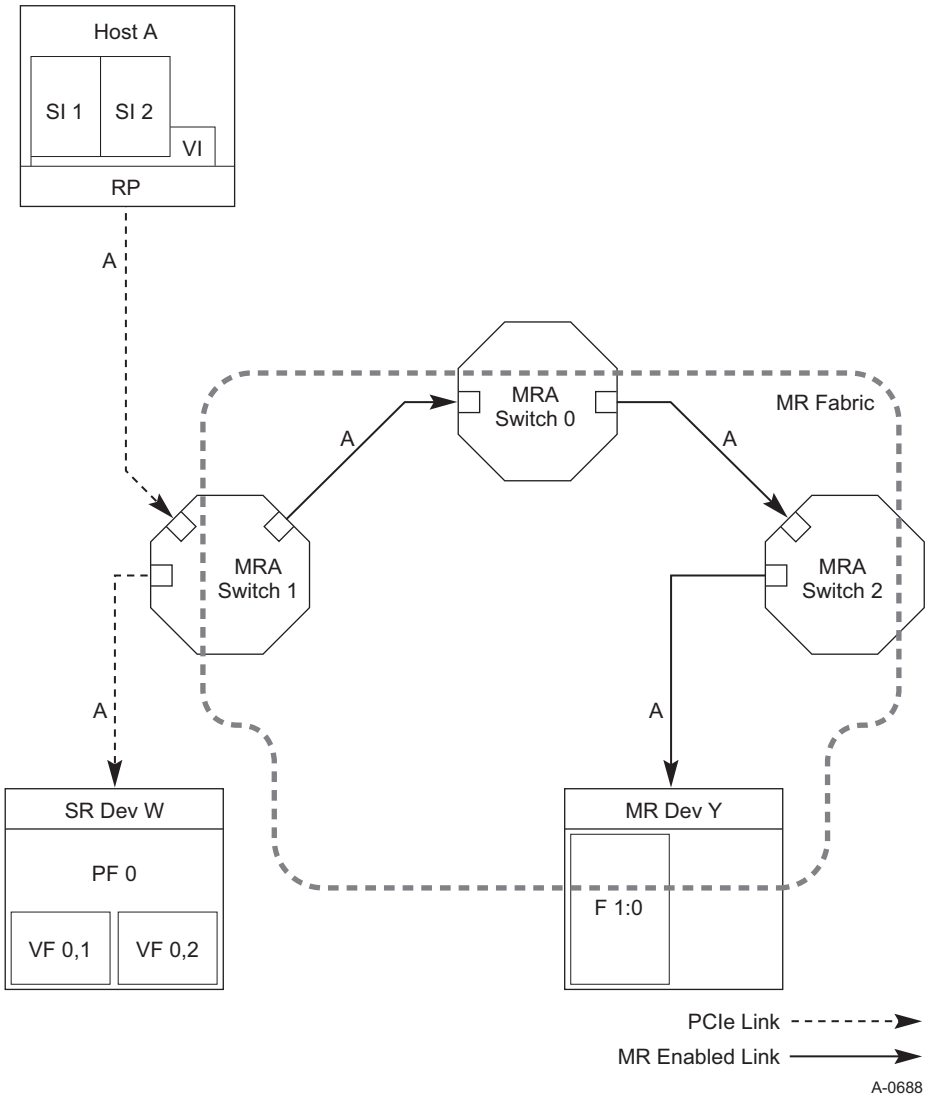

Figure 1-12: Example Multi-Root Topology as Viewed From Host A

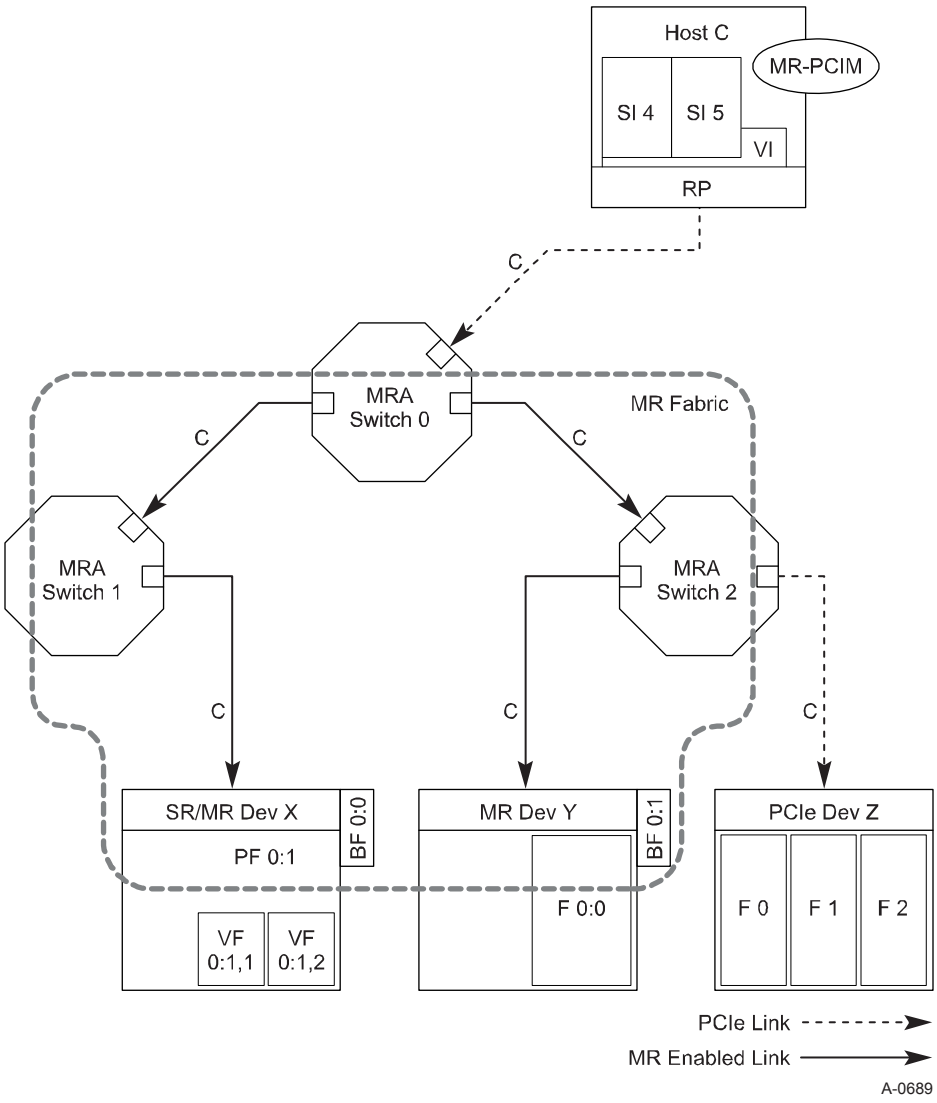

Figure 1-13: Example Multi-Root Topology as Viewed From Host C

2

2. MR Protocol Changes

There are five parts of the PCIe Protocol that are changed to support Multi-Root operation.

- Negotiating use of the MR Link protocol
- Tagging TLPs with a TLP Prefix header
- Supporting per-VH equivalent of Hot Reset
- Supporting enhanced, per-VH flow control
- Processing of certain messages (e.g., INTx, PME)

These will be discussed in the following sections.

2.1. MR Link and Flow Control Negotiation

MR Links use an enhanced Link protocol. For each Link, the use of this enhanced Link protocol is determined by a negotiation between Link partners. This negotiation occurs after the Physical Layer's Link training but before PCIe Flow Control negotiation.

During this negotiation, MR components determine whether their Link Partner agrees to use the MR Link protocol and which version of the Link protocol to use. They also communicate certain MR parameters (MaxVH and MaxVL).

This negotiation occurs by using the new MRInit DLLP as shown in Figure 2-1 and Table 2-1.

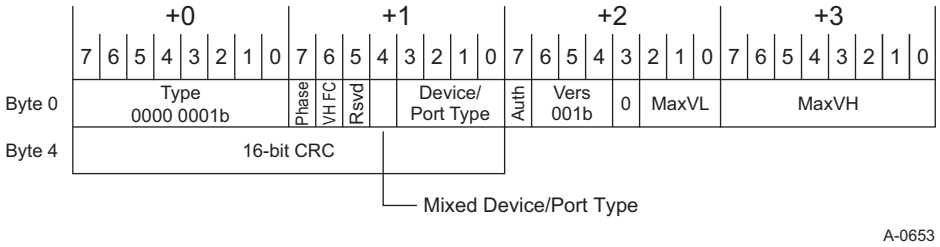

Figure 2-1: MRInit DLLP Format

Table 2-1: MRInit DLLP Fields

Location	Description
Byte 0 Bits 7:0	**Type** – The value 0000 0001b indicates an MRInit DLLP.
Byte 1 Bit 7	**Phase** – Indicates the phase of the MR Negotiation protocol.
Byte 1 Bit 6	**VH FC** – If Set, indicates that the sender supports per-VH, VL flow control. If Clear, indicates that the sender only supports per-VL flow control. Must be set for Switches.
Byte 1 Bit 5	**Reserved**
Byte 1 Bit 4	**Mixed Device/Port Type** – For Ports that only contain Upstream Functions, this bit is Set if VH0 contains multiple Functions that have different values of Device/Port Type in their PCI Express Capability and is otherwise Clear. For Ports that only contain Downstream Functions, this bit is Clear. For Ports that contain a mixture of Upstream and Downstream Functions, this bit is Set. For Ports that, based on mappings, could contain a mixture, this bit is Set even if the current mapping is not mixed.
Byte 1 Bits 3:0	**Device/Port Type** – Device/Port Type of the sender. Encoding is identical to the Device/Port Type field in the PCI Express Capability (Offset 02, Bits 7:4). For Ports that only contain Upstream Functions, the value sent describes Function 0 of VH0. For Ports that only contain Downstream Functions, the value sent reflects the Downstream Function (every VH has exactly one Downstream Function of the same type). For Ports that contain a mixture of Upstream and Downstream Functions, the value sent reflects the Downstream Function. For Ports that, based on mappings, could contain a mixture, this bit reflects the Downstream Function that could be mapped.
Byte 2 Bit 7	**Authorized** – Indicates the sending Port contains at least one Function that is an Authorized Upstream Port on an MR Capable Switch. Must be 0b otherwise. Note: The Authorized Port could be any Function number in any VH.
Byte 2 Bits 6:4	**Protocol Version** – Must be 001b for this version of the specification.
Byte 2 Bit 3	**Reserved** – Transmit 0b.
Byte 2 Bits 2:0	**MaxVL** – Maximum number of VLs minus 1 supported by the sender.
Byte 3	**MaxVH** – Maximum number of VHs minus 1 supported by the sender.

The MRInit DLLP is a new encoding not defined in PCI Express. PCI Express components are required to ignore DLLPs not defined in the *PCI Express Base Specification* (see Section 3.5.2.2 of the *PCI Express Base Specification, Revision. 2.0* or Section 3.5.2.1 of the *PCI Express Base Specification, Revision 1.1*).

MR Devices will always negotiate to use the MR Link protocol. MRA Switches will only negotiate if Link MR-IOV Enable is Set (see Section 4.3.3.2). MRA Root Ports will only negotiate if enabled to do so using vendor specific mechanisms.

A negotiation sequence between two MR components that are enabled to use the Link in MR mode is shown in Figure 2-2.

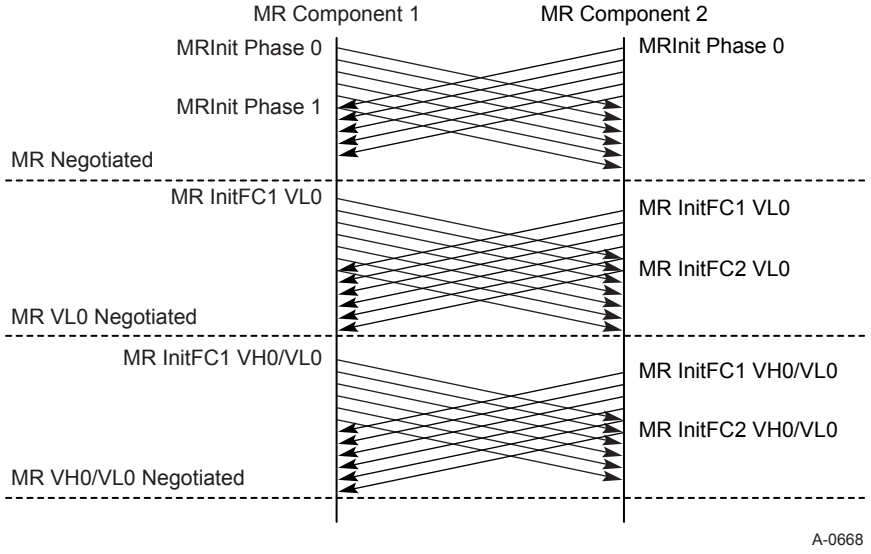

Figure 2-2: Example MR to MR Initialization Sequence

A negotiation sequence between an MR component and a Base component (or an MR component that is not enabled to use MR on the Link) is shown in Figure 2-3.

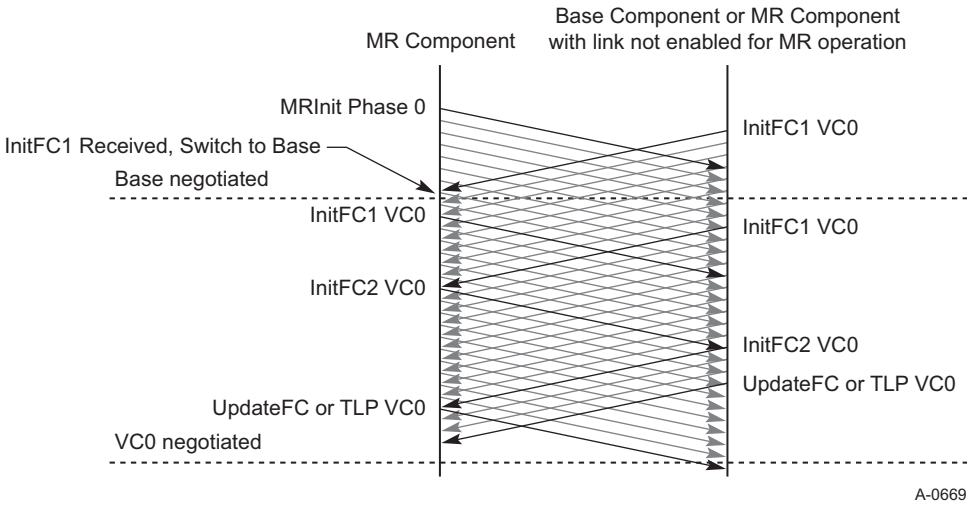

Figure 2-3: Example MR to Base Initialization

The Data Link Control and Management State Machine (DLCMSM) is modified for MR operation. The states for this machine are described below, and are shown in Figure 2-4. Italic text and dashed lines represent Base PCI Express.

States:

- *DL_Inactive* – Physical Layer reporting Link is non-operational or nothing is connected to the Port
- *DL_Init* – Physical Layer reporting Link is operational, initialize Base Flow Control for the default Virtual Channel
- DL_NegotiateMR – Physical Layer reporting Link is operational; negotiate the use of the MR Link Protocol
- DL_InitMR – Physical Link reporting Link is operational; initialize MR Flow Control for VL0 and for VH0/VL0
- *DL_Active* – Normal operation mode

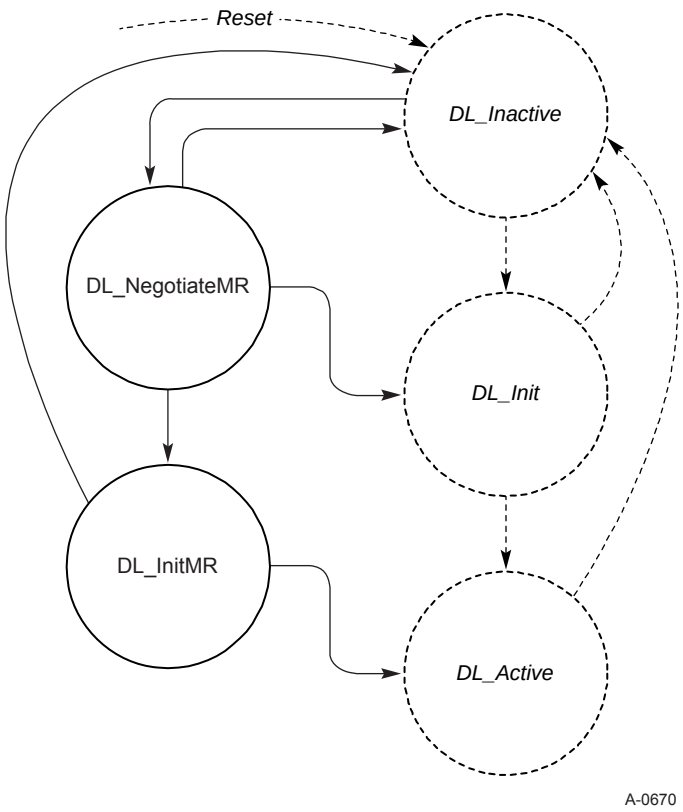

Figure 2-4: MR Data Link Control and Management State Machine (MR-DLCMSM)

The DL_Inactive state rules are modified as follows:

- DL_Inactive

  …

  - Exit to DL_Init if:
    - Indication from the Transaction Layer that the Link is not disabled by software, the Link is not enabled for MR operation, and the Physical Layer reports Physical LinkUp = 1b.
  - Exit to DL_NegotiateMR if:
    - Indication from the Transaction Layer that the Link is not disabled by software, the Link is enabled for MR operation, and the Physical Layer reports Physical LinkUp = 1b.

Rules are added for the new DL_NegotiateMR and DL_InitMR states:

- DL_NegotiateMR
  - While in DL_NegotiateMR:
    - Negotiate MR Link Protocol usage following the MR Link Protocol Negotiation described in Section 2.1.1.
    - Report DL_Down status.

- ♦ The Data Link Layer of a Port with DL_Down status is permitted to discard any received TLPs provided that it does not acknowledge those TLPs by sending one or more Ack DLLPs.

- Exit to DL_Init if:
  - ♦ MR Link Protocol negotiation completes indicating PCIe Link Mode and the Physical Layer continues to report Physical LinkUp = 1b.
- Exit to DL_InitMR if:
  - ♦ MR Link Protocol negotiation completes indicating MR Link Mode and the Physical Layer continues to report Physical LinkUp = 1b.
- Terminate attempt to negotiate MR Link Protocol and Exit to DL_Inactive if:
  - ♦ Physical Layer reports Physical LinkUp = 0b.

❑ DL_InitMR

- While in DL_InitMR:
  - ♦ Initialize Flow Control for the default Virtual Link, VL0, and default Virtual Hierarchy on the default Virtual Link, VH0/VL0, following the Flow Control initialization protocol described in Section 2.1.2.
  - ♦ Report DL_Down status while in state MRFC_INIT1_VL, MRFC_INIT2_VL or MRFC_INIT1_VH; DL_Up status in state MRFC_INIT2_VH.
  - ♦ The Data Link Layer of a Port with DL_Down status is permitted to discard any received TLPs provided that it does not acknowledge those TLPs by sending one or more Ack DLLPs.
- Exit to DL_Active if:
  - ♦ Flow Control initialization completes successfully, and the Physical Layer continues to report Physical LinkUp = 1b
- Terminate attempt to initialize Flow Control and Exit to DL_Inactive if:
  - ♦ Physical Layer reports Physical LinkUp = 0b.

2.1.1. MR Link Protocol Negotiation

The MR Link Protocol Negotiation involves two phases.

❑ Phase 0 entered when MR Link Protocol Negotiation is required.

- Entrance to DL_NegotiateMR state

❑ While in Phase 0 of the MR Link Protocol Negotiation:

- Transmission of TLPs and DLLPs other than the MRInit DLLP must be blocked.
- Continuously transmit an MRInit DLLP as shown in Figure 2-1. MaxVL, MaxVH, Auth and Device/Port Type reflect the sender's values. Protocol Version is 1h. Phase is 0b.

- ♦ This does not block Physical Layer initiated transmissions (for example, Ordered Sets).

- Process received MRInit DLLPs:
  - ♦ Record the MaxVL, MaxVH, VH FC, Device/Port Type, Mixed Device / Port Type and Authorized values.
- Exit to Phase 1 of the MR Link Protocol Negotiation if:
  - ♦ An MRInit DLLP was received with Protocol Version 1h (with either Phase).
- Exit indicating PCIe Link Mode if either:
  - ♦ Any InitFC1 DLLP was received.
  - ♦ Any MRInit DLLP was received with Protocol Version less than 1h (of either Phase).4
- Ignore any received DLLPs other than MRInit and InitFC1. . This includes MRInit DLLPs with a Protocol Version greater than 1h.

❑ While in Phase 1 of the MR Link Protocol:

- Transmission of TLPs and DLLPs other than the MRInit DLLP must be blocked.
- Continuously transmit an MRInit DLLP as shown in Figure 2-1. MaxVL, MaxVH, Auth and Device/Port Type reflect the sender's values. Protocol Version is 1h. Phase is 1b.
  - ♦ This does not block Physical Layer initiated transmissions (for example, Ordered Sets).
- Process received MRInit DLLPs:
  - ♦ Ignore the MaxVL, MaxVH, VF FC, Device/Port Type, and Authorized values.
- Exit indicating MR Link Mode if either:
  - ♦ MRInit DLLP was received with Protocol Version 1h and Phase 1b.
  - ♦ Any MRInitFC1_VL DLLP was received.
- Ignore any received DLLPs other than MRInit and MRInitFC1_VL.

2.1.2. MR Flow Control Initialization Protocol

Before starting normal operation following link training, it is necessary to initialize Flow Control for the default Virtual Link, VL0, and the default Virtual Hierarchy, VH0. In addition, when additional Virtual Links (VLs) and Virtual Hierarchies (VHs) are enabled, the Flow Control initialization process must be completed for each newly enabled VL or VH before it can be used. Flow Control initialization also occurs after a VL or VH is re-enabled, after being disabled and after exiting Reset. This section describes the initialization process that is used for all VLs and all VHs. Note that since VL0 and VH0 are enabled before all other VLs and VHs, no TLP traffic of any kind will be active prior to initialization of VL0 and VH0. However, when additional VLs or VHs are being initialized,

[4] Protocol Version 0h is used by some pre-standard components.

there will typically be TLP traffic flowing on other, already enabled, VLs and VHs. Such traffic has no direct effect on the initialization process for the additional VL(s) and VH(s).

No TLPs may be sent until MR VL/VH initialization has completed on the associated VL and VH.

There are four states in the MR VL/VH initialization process. These states are shown in Figure 2-5.

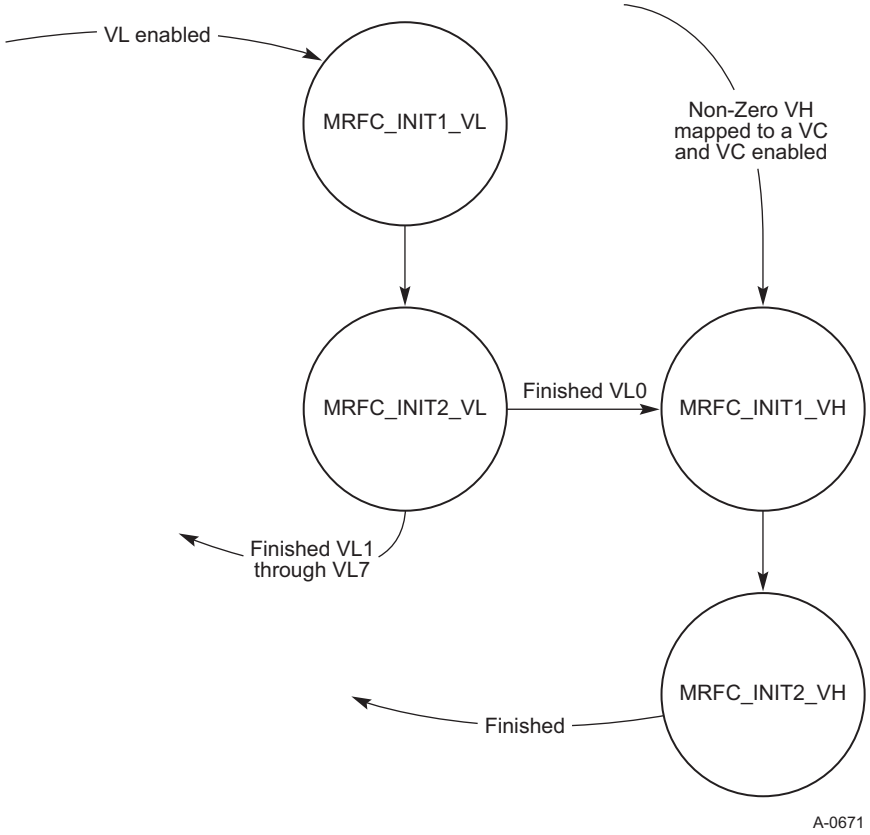

Figure 2-5: MR InitFC State Machine

Conceptually, there is a distinct instance of this state machine for each enabled VH and VL. Each instance of this state machine operates independently and may not block any other instance of this state machine except as described below.

Flow Control Initialization state machines start State MRFC_INIT1_VL is entered when a VL is enabled either automatically (VL0) or explicitly (VL1-7). State MRFC_INIT1_VH is entered either from MRFC_INIT2_VL or because some VH was mapped by MR-PCIM and enabled by software in the VH resulting in a VH for which flow control negotiation has not yet occurred.

The rules for this process are defined in Section 2.1.2.2.

Table 2-2: MR Flow Control Negotiation

Condition	When FC Negotiation Starts:
VL0	MRFC_INIT1_VL for VL0 starts upon entry to DL_InitMR.
(VH0 VL0)	MRFC_INIT1_VH for (VH0 VL0) starts after all of the following are true: 1. MRFC_INIT2_VH has not been successfully completed for (VH0 VL0) since the most recent entry to DL_InitMR or reset within VH0. 2. MRFC_INIT2_VL has successfully completed for VL0. 3. For Switches, when a VS Bridge is mapped to VH0 on this Port.
VLy, where: 1 ≤ y ≤ MaxVL	MRFC_INIT1_VL for VLy starts after all of the following are true: 1. MRFC_INIT2_VL has not been successfully completed for VLy since the most recent entry to DL_InitMR. 2. MR Enable is Set. 3. VL Enable bit for VLy is Set (see Sections 4.2.1.3 and 4.3.3.2).
(VHx VLy), where either: (1 ≤ x ≤ NumVH and 0 ≤ y ≤ MaxVL) or (0 ≤ x ≤ NumVH and 1 ≤ y ≤ MaxVL)	MRFC_INIT1_VH for (VHx VLy) starts after all of the following are true: 1. MRFC_INIT2_VH has not been successfully completed for (VHx VLy) since the most recent entry to DL_InitMR or reset within VHx. 2. MRFC_INIT2_VL has successfully completed for VLy. 3. For Switches, when a VS Bridge is mapped to VHx on this Port. 4. Some entry (VCz) in the VC to VL Map associated with VHx contains the value y in the VCz to VL Map field and has the VCz VL Map Enable bit Set. For Devices, the Function VC to VL Map is used (Section 4.2.4.4). For Switches, the VS Bridge VC to VL Map is used (Section 4.3.6.3). 5. Software running in the VH enables VCz by setting some VC Resource of the VF/MFVC Capability so that VC ID contains z and VC Enable is Set. Note: VC Enable is always set for VC0, and is implicitly set for devices that do not implement the VC or MFVC Capability. 6. MR Enable is Set.
VLy, where: y > MaxVL	Never
(VHx VLy) where: x > NumVH or y > MaxVL	Never

2.1.2.1. MR Flow Control DLLP Encoding

MR Flow Control DLLPs need to communicate the VH number in addition to the Base PCIe information. It is no longer possible to fit this information in a single DLLP. Consequently, for MR-IOV, Header and Data credits are communicated using different DLLPs. The formats for the

various MF Flow Control DLLPs are shown in Figure 2-6 through Figure 2-15. The DLLP fields are described in Table 2-3. If a Component advertises infinite VH credits, then the Component must transmit PCIe Base UpdateFC DLLPs for that VL instead of the MRUpdateFC DLLPs described below.

- The VL Credit Type is implicitly determined from the DLLP Type Encoding used by the UpdateFC DLLP.
- The VC ID field in the UpdateFC DLLP contains the VL number.
- The HdrFC field in the UpdateFC DLLP contains VL header credit value for the indicated type (P, NP, or Cpl).
- The DataFC field in the UpdateFC DLLP contains the VL data credit value for the indicated type (P, NP, or Cpl).

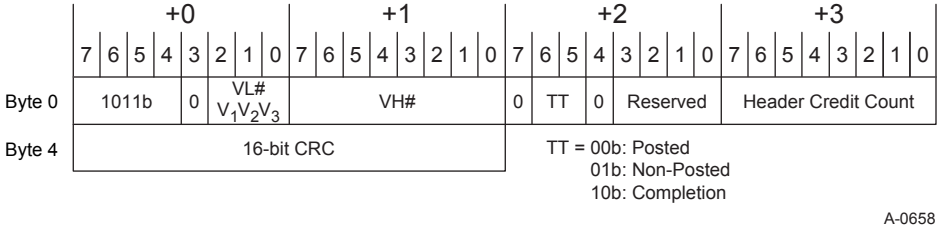

Figure 2-6: MRUpdateFC Header DLLP

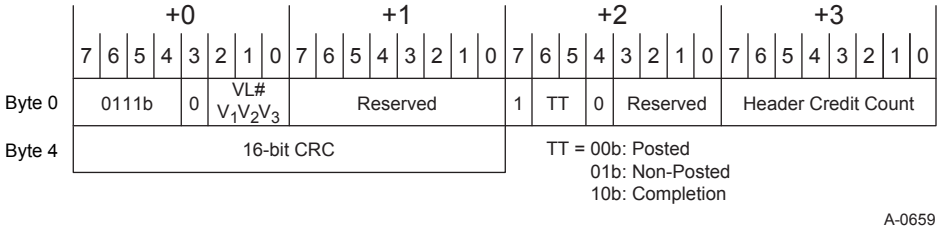

Figure 2-7: MRInitFC1_VL Header DLLP

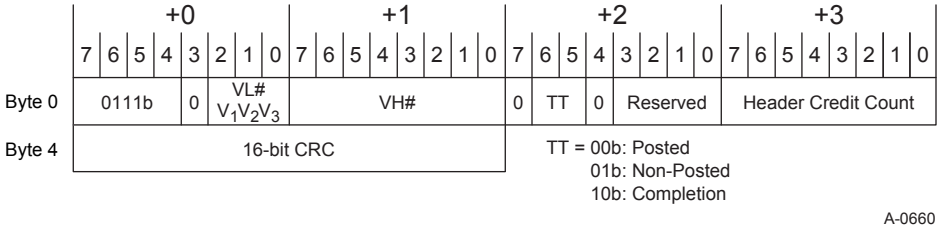

Figure 2-8: MRInitFC1_VH Header DLLP

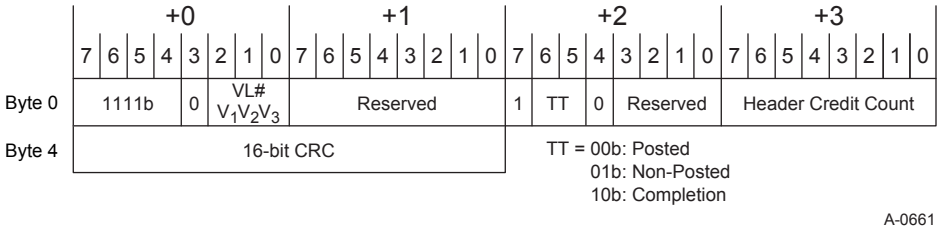

Figure 2-9: MRInitFC2_VL Header DLLP

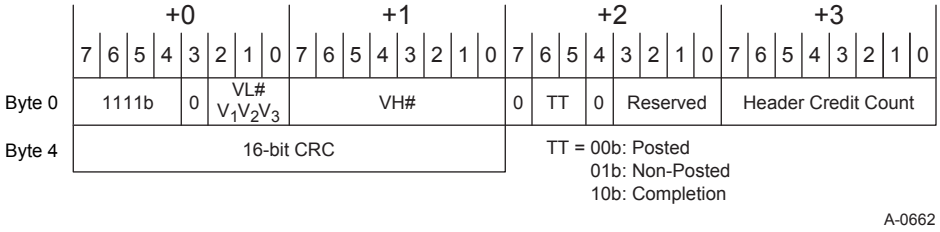

Figure 2-10: MRInitFC2_VH Header DLLP

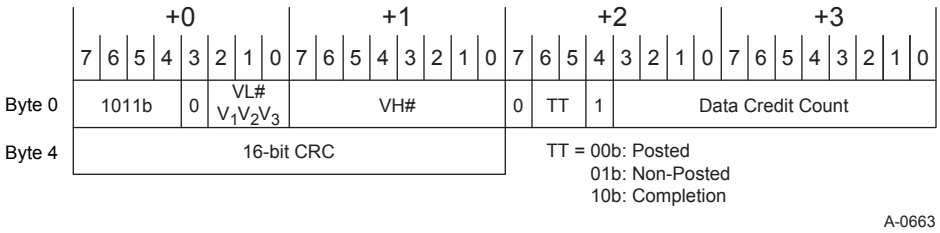

Figure 2-11: MRUpdateFC Data DLLP

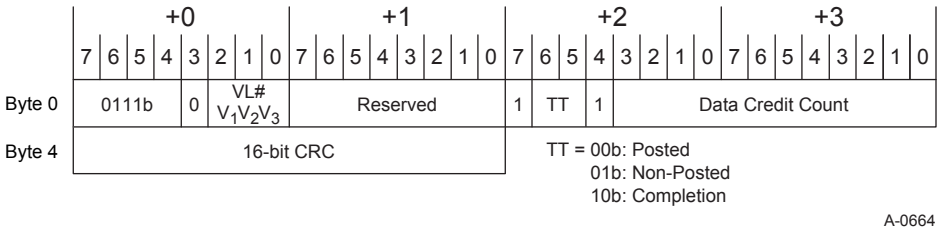

Figure 2-12: MRInitFC1_VL Data DLLP

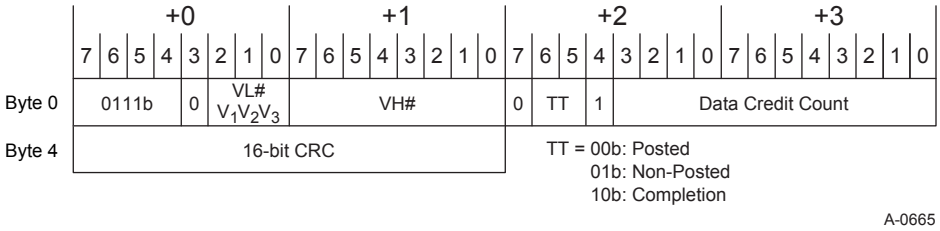

Figure 2-13: MRInitFC1_VH Data DLLP

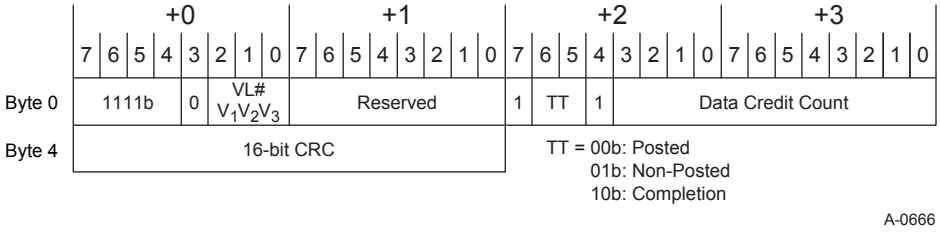

Figure 2-14: MRInitFC2_VL Data DLLP

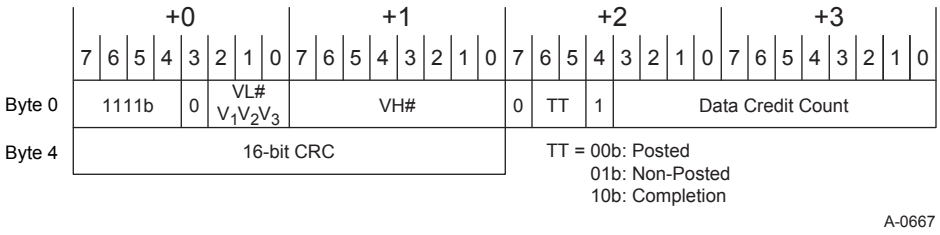

Figure 2-15: MRInitFC2_VH Data DLLP

Table 2-3: MR Flow Control DLLP Fields

Location	Description
Byte 0 Bits 7:4	**DLLP Type** – 1011b indicates an MRUpdateFC DLLP 0111b indicates an MRInitFC1_VL or MRInitFC1_VH DLLP 1111b indicates an MRInitFC2_VL or MRInitFC2_VH DLLP
Byte 0 Bits 2:0	**VL Number** – Indicates the Virtual Link
Byte 1	**VH Number** – Indicates the Virtual Hierarchy. This field is reserved if VH Omitted is Set.
Byte 2 Bit 7	**VH Omitted** – Indicates whether the VH Number field is present in the DLLP. If Set, this indicates the DLLP is MRInitFC1_VL or MRInitFC2_VL and the VH Number is omitted.
Byte 2 Bits 6:5	**TT (TLP Type)** – 00b indicates Posted credit 01b indicates Non-Posted credit 10b indicates Completion credit 11b is Reserved
Byte 3 Bit 4	**Credit Type** – 0 indicates Header Credit 1 indicates Data Credit
Byte 3 Bits 3:0 and Byte 4	**Data Credit Value** – If Credit Type is Set, PCIe encoding applies (i.e., during initialization, zero means infinite).
Byte 4	**Header Credit Value** – If Credit Type is Clear, PCIe encoding applies (i.e., during initialization, zero means infinite).

2.1.2.2. MR Flow Control Initialization State Machine Rules

- If at any time during initialization for VLs 1-7 a VL is disabled, any flow control initialization process involving that VL is terminated.[5]
- If at any time during initialization for (VHx , VLy), VHx exits either the US VH Up or DS VH Up reset states, any flow control initialization process involving (VHx VLy) is terminated (see Section 2.3.3).
- If at any time during initialization for (VHx , VLy), VL Map Enables are cleared so that no VC is mapped to (VHx VLy), any flow control initialization process involving (VHx VLy) is terminated.
- Rules for state MRFC_INIT1_VL:
  - Entered when initialization of a VL (VLx) is required.
    - Entrance to DL_InitMR state (VLx = VL0)
    - When a MR Enable is Set and a VL (VLx = VL1-7) is enabled by software (see Sections 4.2.1.3 and 4.3.3.2)

[5] This includes both VL and (VH VL) flow control negotiation.

- While in MRFC_INIT1_VL:
  - Transaction Layer must block transmission of TLPs using VLx.
  - Transmit the following six MRInitFC1_VL DLLPs for VLx in the following relative order:
    - MRInitFC1_VL – P – Header (first)
    - MRInitFC1_VL – P – Data (second)
    - MRInitFC1_VL – NP – Header (third)
    - MRInitFC1_VL – NP – Data (fourth)
    - MRInitFC1_VL – Cpl – Header (fifth)
    - MRInitFC1_VL – Cpl – Data (sixth)
  - The six MRInitFC1_VL DLLPs must be transmitted at least once every 34 µs.
    - Time spent in the Recovery LTSSM state does not contribute to this limit.
    - It is strongly encouraged that the MRInitFC1_VL DLLP transmissions are repeated frequently, particularly when there are no other TLPs or DLLPs available for transmission.
  - Except as needed to ensure at least the required frequency of MRInitFC1_VL DLLP transmission, the Data Link Layer must not block other transmissions.
    - Note that this includes all Physical Layer initiated transmissions (for example, Ordered Sets), Ack and Nak DLLPs (when applicable), and TLPs using VLs and VHs that have previously completed initialization (when applicable).
  - Process received MRInitFC1_VL and MRInitFC2_VL DLLPs for VLx:
    - Record the indicated FC unit values.
    - Set Flag FI1 once FC unit values have been recorded for each of P_Hdr, P_Data, NP_Hdr, NP_Data, Cpl_Hdr, and Cpl_Data of VLx.
- Exit to MRFC_INIT2_VL if:
  - Flag FI1 has been set indicating that FC unit values have been recorded for each of P_Hdr, P_Data, NP_Hdr, NP_Data, Cpl_Hdr, and Cpl_Data of VLx.

❑ Rules for state MRFC_INIT2_VL:

- While in MRFC_INIT2_VL:
  - Transaction Layer must block transmission of TLPs using VLx.
  - Transmit the following six MRInitFC2 DLLPs for VLx in the following relative order:
    - MRInitFC2_VL – P – Header (first)
    - MRInitFC2_VL – P – Data (second)
    - MRInitFC2_VL – NP – Header (third)
    - MRInitFC2_VL – NP – Data (fourth)

- MRInitFC2_VL – Cpl – Header (fifth)
- MRInitFC2_VL – Cpl – Data (sixth)

♦ The six MRInitFC2_VL DLLPs must be transmitted at least once every 34 µs.
- Time spent in the Recovery LTSSM state does not contribute to this limit.
- It is strongly encouraged that the MRInitFC2_VL DLLP transmissions are repeated frequently, particularly when there are no other TLPs or DLLPs available for transmission.

♦ Except as needed to ensure at least the required frequency of MRInitFC2 DLLP transmission, the Data Link Layer must not block other transmissions.
- Note that this includes all Physical Layer initiated transmissions (for example, Ordered Sets), Ack and Nak DLLPs (when applicable), and TLPs using VLs and VHs that have previously completed initialization (when applicable).

♦ Process received MRInitFC1_VL and MRInitFC2_VL DLLPs for VLx:
- Ignore the indicated FC unit values.
- Set flag FI2 on receipt of any MRInitFC2_VL DLLP for VLx.

♦ Set flag FI2 on receipt of any MRInitFC1_VH DLLP for any VH on VLx.

- Signal completion and exit if:
  ♦ Flag FI2 has been set.
  ♦ Note: If VL0 was being initialized, the next state will be MRFC_INIT1_VH because VH/VL initialization is needed for (VH0 VL0).

❑ Rules for state MRFC_INIT1_VH:

- Entered when initialization of a VH/VL (VHx VLy) is required. This occurs when all of the following conditions are true:
  ♦ MRFC_INIT1_VH has not been entered for (VHx VLy) since entry to DL_InitMR, since VL Enable for VLy was Set, or since VHx has exited reset.
  ♦ MRFC_INIT2_VL has been completed for VLy since entry to DL_InitMR or since VL Enable for VLy has been Set.
  ♦ Either MR Enable is Set or (VHx VLy) is (VH0 VL0).
  ♦ For some VCz, VCz VL Map Enable is Set, and VCz VL Map contains VLy.
  ♦ VCz is enabled within the VH as defined in the *PCI Express Base Specification* (e.g., because VC Enable for VCz is Set in the appropriate VC/MFVC Capability, or because VCz is VC0 and VC/MFVC Capabilities are not present).
  ♦ No MRUpdateFC DLLPs for (VHx VLy) are scheduled for transmission.
- If the Link partner indicated it does not support per-(VH VL) flow control during DL_NegotiateMR (i.e., it sent MRInit DLLPs with the VH FC bit Clear), then this component must transmit infinite VH credits in all MRInitFC1_VH DLLPs sent for all VHs and all Credit Types.

- While in MRFC_INIT1_VH:
  - Transaction Layer must block transmission of TLPs using VHx VLy.
  - Transmit the following six MRInitFC1_VH DLLPs for (VHx VLy) in the following relative order:
    - MRInitFC1_VH – P – Header (first)
    - MRInitFC1_VH – P – Data (second)
    - MRInitFC1_VH – NP – Header (third)
    - MRInitFC1_VH – NP – Data (fourth)
    - MRInitFC1_VH – Cpl – Header (fifth)
    - MRInitFC1_VH – Cpl – Data (sixth)
  - The six MRInitFC1_VH DLLPs must be transmitted at least once every 34 µs.
    - Time spent in the Recovery LTSSM state does not contribute to this limit.
    - It is strongly encouraged that the MRInitFC1_VH DLLP transmissions are repeated frequently, particularly when there are no other TLPs or DLLPs available for transmission.
  - Except as needed to ensure at least the required frequency of MRInitFC1_VH DLLP transmission, the Data Link Layer must not block other transmissions.
    - Note that this includes all Physical Layer initiated transmissions (for example, Ordered Sets), Ack and Nak DLLPs (when applicable), and TLPs using VLs and VHs that have previously completed initialization (when applicable).
  - Process received MRInitFC1_VH and MRInitFC2_VH DLLPs for (VHx VLy):
    - Record the indicated FC unit values.
    - Set Flag FI3 once FC unit values have been recorded for each of P_Hdr, P_Data, NP_Hdr, NP_Data, Cpl_Hdr, and Cpl_Data of (VHx VLy).
- Exit to MRFC_INIT2_VH if:
  - Flag FI3 has been set indicating that FC unit values have been recorded for each of P_Hdr, P_Data, NP_Hdr, NP_Data, Cpl_Hdr, and Cpl_Data of (VHx VLy).

❑ Rules for state MRFC_INIT2_VH:

- While in MRFC_INIT2_VH:
  - Transaction Layer must block transmission of TLPs using (VHx VLy).
  - If the Link partner indicated it does not support per-(VH VL) flow control during DL_NegotiateMR (i.e., it sent MRInit DLLPs with the VH VL bit Clear), then this component must transmit infinite VH credits in all MRInitFC2_VH DLLPs sent for all VHs and all Credit Types.
  - Transmit the following six MRInitFC2_VH DLLPs for VHx VLy in the following relative order:

- MRInitFC2_VH – P – Header (first)
- MRInitFC2_VH – P – Data (second)
- MRInitFC2_VH – NP – Header (third)
- MRInitFC2_VH – NP – Data (fourth)
- MRInitFC2_VH – Cpl – Header (fifth)
- MRInitFC2_VH – Cpl – Data (sixth)

♦ The six MRInitFC2_VH DLLPs must be transmitted at least once every 34 µs.
- Time spent in the Recovery LTSSM state does not contribute to this limit.
- It is strongly encouraged that the MRInitFC2_VH DLLP transmissions are repeated frequently, particularly when there are no other TLPs or DLLPs available for transmission.

♦ Except as needed to ensure at least the required frequency of MRInitFC2_VH DLLP transmission, the Data Link Layer must not block other transmissions.
- Note that this includes all Physical Layer initiated transmissions (for example, Ordered Sets), Ack and Nak DLLPs (when applicable), and TLPs using VLs and VHs that have previously completed initialization (when applicable).

♦ Process received MRInitFC1_VH and MRInitFC2_VH DLLPs for (VHx VLy):
- Ignore the indicated FC unit values.
- Set flag FI4 on receipt of any MRInitFC2_VH DLLP for (VHx VLy).

♦ Set flag FI4 on receipt of any TLP on (VHx VLy), or any MRUpdateFC DLLP for (VHx VLy).[6]

- Signal completion and exit if:
  ♦ Flag FI4 has been set.

2.2. TLP Prefix Tagging

After successful MR Link Protocol negotiation, links use the MR Enhanced Link Protocol. TLPs on such links contain a TLP Prefix as shown in Figure 2-16. This prefix is located between the Sequence Number and the PCIe TLP header.

[6] Note: UpdateFC will not set FI4. See Section 2.4.1.

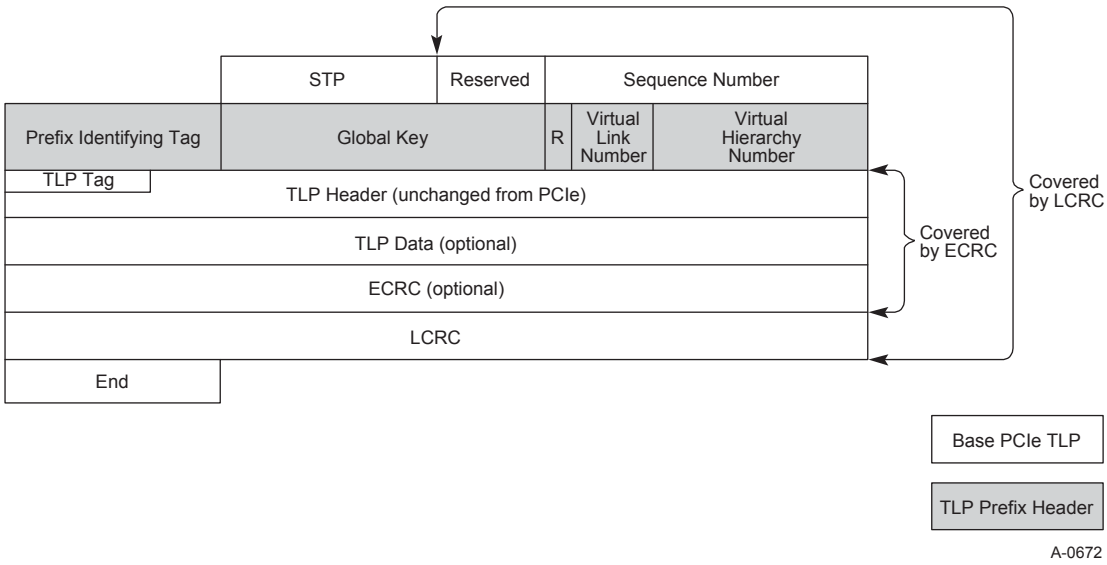

Figure 2-16: TLP Prefix Header Location

The TLP Prefix is part of the wire packet. It is covered by LCRC and is retransmitted with the rest of the TLP under the same rules. Sequence Numbers and the Ack/Nak protocol remain a per-Link notion and are not affected by Multi-Root operation.

The TLP header, Data, and ECRC are not changed from PCIe TLP usage.

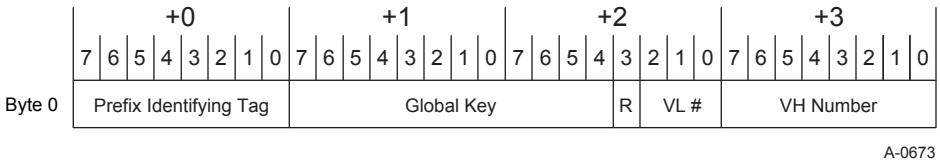

Figure 2-17: TLP Prefix Header Layout

The layout of the TLP Prefix header is shown in Figure 2-17. The first byte is a fixed Prefix Identifying Tag value. The value of the Prefix Identifying Tag is defined in a ECN to the PCI Express Base Specification that is currently under discussion within the Protocol Working Group. This value will be determined prior to release of the 1.0 version of this specification. This value is not used by any TLP so that the presence of the TLP Prefix can be determined by examining the TLP in isolation without needing additional Link-state information. Doing so allows test equipment to better understand things without needing to observe the initialization sequence.

Virtual Hierarchy Numbers (VHN) are Link-local. On a Link that supports *n* VHs, valid VHNs are [0 .. *n*-1]. A given TLP may take on different VHN values on each Multi-Root Link that it traverses.

The VL # field contains the Virtual Link Number (VL #). It contains information used to support Congestion Management and Isolation. TLPs are assigned to VLs using a combination of PCIe TC to VC mapping rules and new MR VC to VL mapping rules. See Chapter 8 for details.

The Global Key field is used to guard against "VH Hopping." VH Hopping is when a TLP in one VH inadvertently ends up on a different VH (either due to MR-PCIM table configuration errors or due to a hardware error inside an MR Switch or MR Device). The Global Key is added to the TLP

when the TLP Prefix is attached at the MR Ingress point. The Global Key value selected is based on which VH is being used. This Global Key value is preserved, unchanged, through subsequent MR Switches. The Global Key value may be validated against the expected value at various points in the MR topology. Validation of the Global Key value at MR Egress can be enabled by software. Validation of the Global Key value at either (or both) MR Switch Input or MR Switch Output is optional and, if supported, can be enabled by software. Global Key checking is similar to PCIe ECRC checking. Failure of any Global Key validation is an unrecoverable error.

MR-PCIM software configures the tables used to generate and check the Global Key values. To avoid Global Key mismatch errors, MR-PCIM must configure tables such that all TLPs in a given VH have the same Global Key value. To provide maximum protection, MR-PCIM should configure tables so that TLPs in different VHs have different Global Keys (this protection is not provided between VHs with duplicate Global Key values).

2.2.1. MR Switch Transaction Layer Processing

MR Switches implement a set of Virtual Switches (VS). Each VS is assigned by software to a single Virtual Hierarchy that, in turn, is associated with a single Root Port. Within a Switch, TLPs are associated with a VS and are routed within that VS using PCIe routing and ordering rules (e.g., address routed, ID routed, broadcast to/from root). TLPs for unrelated VHs are unordered.

Conceptually, TLPs are processed by a Switch as follows:

1. TLPs arrive on an Input Port of the Switch with a Link-local Input VH Number.
   a. For a Link operating in MR mode, the Input VHN is contained in the TLP Prefix header.
   b. For a Link operating in Base PCIe mode, the Input VHN is 0.
2. Switch mapping tables are used to map the incoming TLP to a VS and a specific PCI-to-PCI Bridge of that VS.

   f(Input Port, Input VHN) → {VS, Input Bridge}

   If no mapping exists, the incoming TLP is discarded and an MR Uncorrectable Non-Fatal TLP Error is signaled.
3. Global Key input processing occurs.
   a. If the input Link was operating in MR mode, the Global Key from the TLP is optionally validated against the Global Key associated with the VS. This is the "entering check" as described in Section 4.3.5.2.
   b. If the input Link was operating in Base PCIe mode, there was no Global Key to check against. This is the MR Ingress point for this TLP and the TLP is assigned the Global Key associated with the VS.
4. The TLP is routed within the VS. This mapping uses conventional PCIe rules using mapping tables controlled by software operating in the VH. If the TLP is consumed by the VS, these rules indicate routing within the VS (i.e., which Type 1 header, etc.). If the TLP is being forwarded onward, these rules select an Output Bridge and an output Virtual Channel (VC).

   g(Input Bridge, Base TLP header) → {Type 1 Header …} *if TLP is local to the VS*
   → {Output Bridge, VC} *if TLP is being forwarded*

5. For TLPs being consumed by the VS or being forwarded to an Output Port operating in Base PCIe mode, this Switch is the MR Egress point for the TLP, and Global Key checking occurs. The Global Key associated with the TLP is optionally validated against the Global Key associated with the VS. This is the "terminating check" as described in Section 4.3.5.2.
6. For TLPs being forwarded, the Switch mapping tables are used to map the outgoing TLP to an Output Port, an Output VHN of that Port and an Output Virtual Link (VL) of that Port

   h(VS, Output Bridge, VC) → {Output Port, Output VHN, Output VL}
7. The VL and (VH VL) Flow Control gates are checked using the Output Port's values to verify that there are sufficient credits to forward the TLP.
8. Port Arbitration is performed using PCIe rules within the VS. Each Downstream Bridge is considered a distinct Port for this purpose.
9. For an output Link operating in MR mode, VH Arbitration occurs within the Output Port, Output VL. Arbitration scheme used is fixed round robin.
10. For an output Link operating in MR mode, VL Arbitration occurs within the Output Port. Arbitration scheme is controlled by the VL Arbitration information in the Port Table.
11. For an output Link operating in Base PCIe mode, VC Arbitration occurs within the Output Port. Arbitration scheme is controlled by software using VC Arbitration information associated with the Type 1 header within the VS. Programmable VC Arbitration is optional in PCIe and remains optional for links operating in Base PCIe mode. VC Arbitration is not supported for links operating in MR mode.
12. The Output Port forwards the TLP.
    a. For an output Link operating in Base PCIe mode, the Output VHN is zero. The Virtual Channel (VC) used to transmit the TLP was determined in step 4 above. The Output VL value is not used.
    b. For an output Link operating in MR mode, new Output VHN and Output VL values are placed in the TLP Prefix header overwriting the input values (if any). The Output VL is used to transmit the TLP.
13. Global Key output processing occurs. If the output Link is operating in MR mode, the Global Key from the TLP is optionally validated against the global key associated with the VF. This is the "exiting check" as described in Section 4.3.5.2. If the output Link is operating in Base PCIe mode, the "terminating check" in step 5 is used instead.
14. When receive buffer space is made available, Flow Control is returned to the VL and (VH VL) contained in the TLP Prefix. The TC to VC maps and VC to VL maps on the receiver are not used (PCIe does not transmit the VC so the TC to VC map is used for this purpose).

2.2.2. MR Device Transaction Layer Processing

MR Devices implement a collection of Functions in each VH. VH0 is used to manage the device. Within a Device, TLPs are associated with a VH and are routed within that VH using PCIe routing and ordering rules. TLPs for unrelated VHs are unordered.

2.2.2.1. Receiving TLPs

TLPs received by an MR Device are processed as follows:

1. TLPs arrive from the MR Switch with a Link-local Input VH Number contained in the TLP Prefix header.
2. The Global Key from the TLP is validated against the global key associated with the VH (see Section 2.2.3).
3. The addressed Function within the VH is determined using rules defined in the *PCI Express Base Specification* and the *Single-Root I/O Virtualization and Sharing Specification*.
4. The addressed Function is one of PF, VF, Function, or BF. These are designated as **PF *h:f,*** **VF *h:f,s***, **F *h:f,*** or ***BF bf***, respectively (using the nomenclature described in Section 1.2).
5. TC checking occurs as described in the *PCI Express Base Specification*. As described in the *Single Root I/O Virtualization and Sharing Specification*, the VC and/or MFVC Capabilities of the PF are used for VFs.
6. The VH number ***h*** and PF/VF/Function number ***f*** determine the associated BF and corresponding Function Table Entry of that BF. For unmanaged Functions in VH0, there will not be an entry.
7. For Functions, BFs, and PFs, a Vendor Specific mechanism determines the underlying Function.
8. For VFs, VF Mapping is used to determine the underlying Function.
    a. If the BF located in step 6 supports VF Mapping, the MVF number is contained in the LVF Table Entry located using the VF # together with the TotalVFs and BaseLVF values from the Function Table Entry. This MVF number describes the underlying Function (see Section 3.2.3).
    b. In addition, if VF Migration is supported and enabled, the VF State in the LVF Table Entry must also be checked to ensure the mapping is in either of the Active.Available or Active.MigrateOut states (see Section 3.2.4).
    c. If the BF located in step 6 does not support VF Mapping, a Vendor Specific Mechanism determines the underlying Function.
9. The TLP is handed to underlying Function for processing.
10. When receive buffer space is made available, Flow Control is returned to the VL and (VH VL) contained in the TLP Prefix. The TC to VC maps and VC to VL maps on the receiver are not used. (In the *PCI Express Base Specification*, the VC is not transmitted and the TC to VC map at the receiver is used to determine the VC that should be updated.) See Section 2.4 for details.

2.2.2.2. Transmitting TLPs

TLPs transmitted by an MR Device are processed as follows:

1. The underlying Function, TC, TLP size, and TLP type (P, NP, Cpl) are determined.
2. For Functions, BFs, and PFs, a Vendor Specific mechanism determines the VH and Function #.

3. For VFs, if VF Mapping is used, a reverse VF map is used to determine the VH and Function #. Otherwise, a Vendor Specific mechanism determines the VH and Function # (see Section 3.2.3).
4. Initial TC to VCID mapping and VC Arbitration occurs. The Mapped Function's VC Capability is used to convert TC to VCID. As described in the *Single Root I/O Virtualization and Sharing Specification*, for VFs, the VC/MFVC Capability of the PF is used.
5. If Function 0 of the VH supports the MFVC Capability, MFVC Arbitration and another round of TC to VCID mapping occurs.
6. VCID to VL mapping occurs using the map contained in the Function Table Entry associated with the Function or PF.
7. Now that the VL is known, VL and (VH VL) Flow Control gates checking occurs.
8. VH Arbitration occurs with the VL. The arbitration scheme is fixed round-robin.
9. VL Arbitration occurs. The VL Arbitration tables are contained in one of the BFs.
10. The TLP is sent on the wire. CREDITS_CONSUMED is updated to reflect this.

2.2.3. Global Key Processing

Global Keys are processed at various points within an MR topology:

- Global Keys are assigned at the MR Ingress point of the TLP. For TLPs originated by an MR Component, the MR Ingress point is inside that originating MR Component. For TLPs originated by a Base PCIe Component, the MR Ingress point is the input Port of an MR Switch.
- Global Keys are checked at the MR Egress point of the TLP. For TLPs destined for an MR Component, the MR Egress point is inside the destination MR Component. For TLPs destined for a Base PCIe Component, the MR Egress point is the output Port of an MR Switch.
- MR Switches can optionally check Global Keys as TLPs enter the Switch from an MR Link.
- MR Switches can optionally check Global Keys as TLPs exit the Switch on an MR Link.

Hardware support for the check as TLPs enter and exit MR Switches is optional. Hardware support for the MR Egress check is mandatory.

The Global key value is a 12-bit value, assigned by software to each VH. To achieve maximum protection, software should assign each VH a distinct Global Key value.

A Global Key check passes if the value in the TLP and the expected value match. A Global Key check also passes if either the expected value is 000h or the TLP value is 000h (the wild card value).

In other words, when MR Ingress points assign a TLP the Global Key value of 000h, Global Key checks will always pass for that TLP. When a Global Key is programmed with an expected value of 000h, any checks that use that value will always pass.

Global Key checking is disabled by default. This allows software time to program the Global Key registers before enabling checking.

2.2.4. MR TLP Dataflow Examples

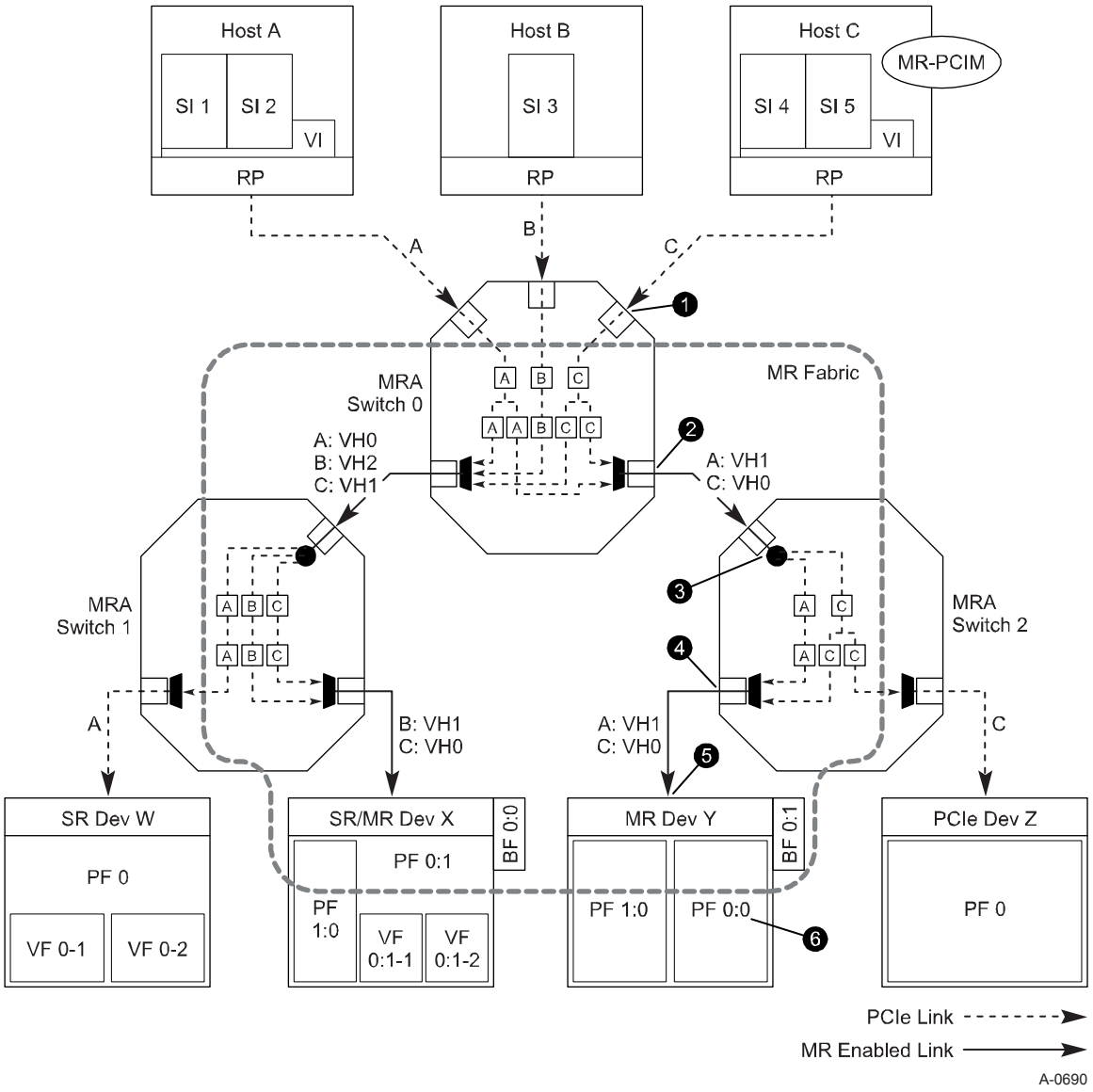

Figure 2-18: MR Dataflow Examples

Consider a Memory Read TLP initiated by SI 4 targeting PF 1:0 in EP Y:

1. The Host C to MRA Switch 0 Link operated in PCIe mode. There is no TLP Prefix.
2. MRA Switch 0 has been programmed so that all TLPs arriving at Port ❶ are associated with the VS for VH C.
3. The VS inside MRA Switch 0 address routes the TLP to a Virtual Downstream Port associated with physical Port ❷ (the Link headed towards MRA Switch 2). MR-PCIM has assigned VH 0 to this Virtual Downstream Port. The TLP exits MRA Switch 0 with a TLP Prefix having VH Number 0.

4. The TLP arrives at MRA Switch 2 Port ❸ labeled as belonging to VH 0. MRA Switch 2 has been programmed so that all TLPs arriving at Port ❸ labeled VH 0 are associated with the VS for VH C.

5. The VS inside MRA Switch 2 address routes the TLP to a Virtual Downstream Port associated with physical Port ❹ (the Link headed towards Device Y). MR-PCIM has assigned VH 1 to this Virtual Downstream Port. The TLP exits MRA Switch 2 with a TLP Prefix having VH Number 1.

6. The TLP arrives at Device Y labeled as belonging to VH 1. The Device hands the transaction to PF 1:0 ❻ for execution.

7. PF 1:0 completes the transaction and emits a completion TLP. Device Y sends this TLP out Port ❺ labeled with VH 1.

8. The TLP arrives at MRA Switch 2 Port ❹ labeled as belonging to VH 1. MRA Switch 2 has been programmed so that all TLPs arriving at Port ❹ labeled VH 1 are associated with the VS for VH C.

9. The VS inside MRA Switch 2 ID routes the TLP to a Virtual Upstream Port associated with physical Port ❸ (the Link headed towards MRA Switch 0). MR-PCIM has assigned VH 0 to this Virtual Upstream Port. The TLP exits MRA Switch 2 with a TLP Prefix having VH Number 0.

10. The TLP arrives at MRA Switch 0 Port ❷ labeled as belonging to VH 0. MRA Switch 0 has been programmed so that all TLPs arriving at Port ❷ labeled VH 0 are associated with the VS for VH C.

11. The VS inside MRA Switch 0 ID routes the TLP to a Virtual Upstream Port associated with physical Port ❶ (the Link headed towards Host C). MR-PCIM has designated this Link a PCIe Link and has assigned this Virtual Upstream Port to it. The TLP exits Switch 0 with a no TLP Prefix.

12. SI 4 in Host C sees a completion for the Memory Read Transaction.

2.3. Per-VH RESET

In Multi-Root, Reset DLLPs are used to implement a per-VH equivalent of PCIe Hot Reset. DLLPs are used to ensure "quick" Reset propagation by avoiding delays introduced by TLP ordering and flow control rules.

Reset of a VH requires discarding TLPs associated with that VH. TLPs associated with VHs not in reset must not be affected. This discard process may take more time that the PCIe equivalent. For example, only some of TLPs in the retry buffer might be affected.

Reset DLLPs supports an acknowledgment protocol to mimic the TS1/TS2 ACK behavior of PCIe. This allows an upstream component to know when a downstream component has entered reset.

2.3.1. Per-VH Reset Example

Figure 2-19 shows an example MR Topology. Figure 2-20 shows Host A's view of the same topology. Table 2-4 describes the events and associated actions involved in propagating a Reset from Host A.

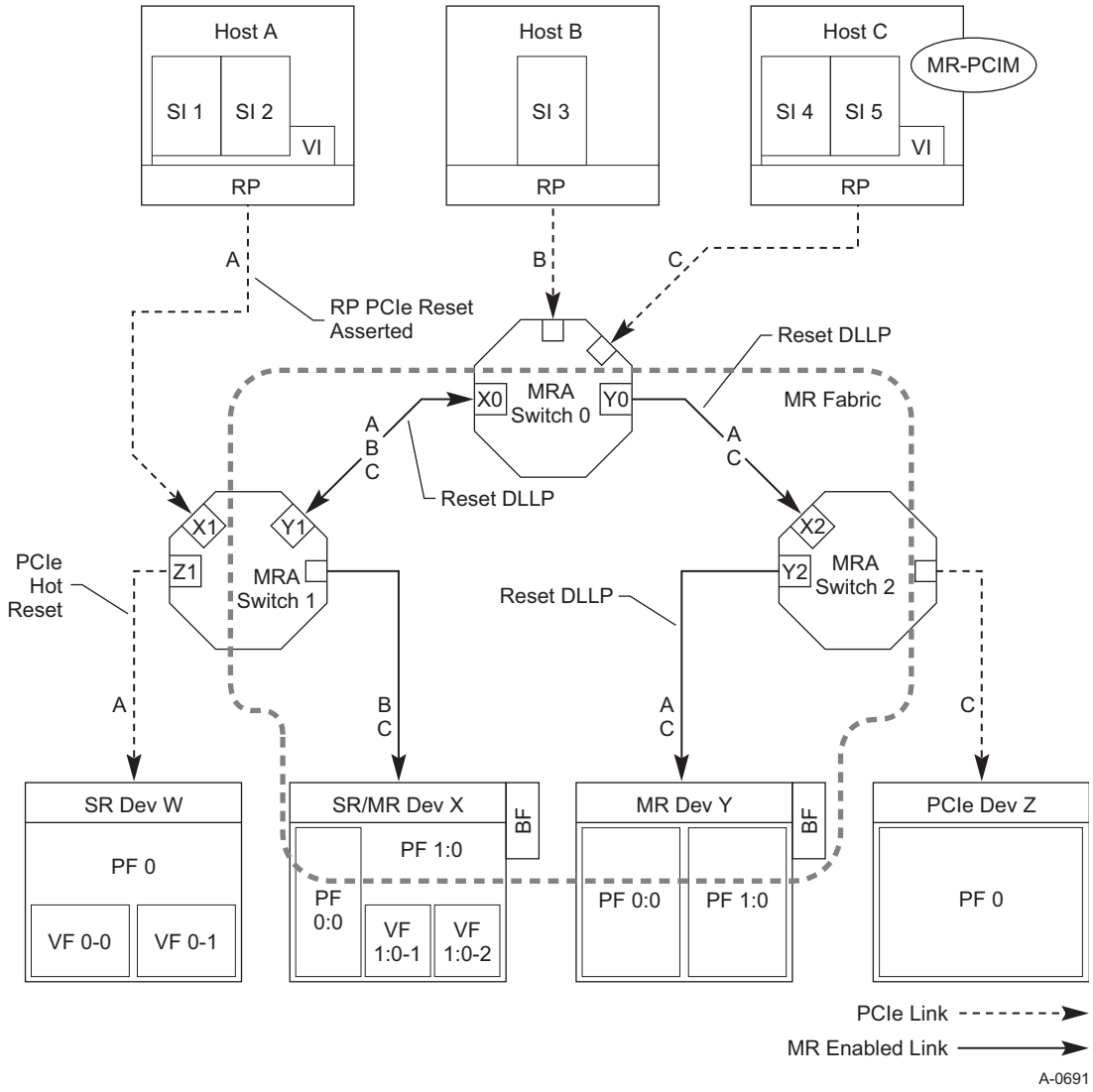

Figure 2-19: Reset DLLP Example: Topology

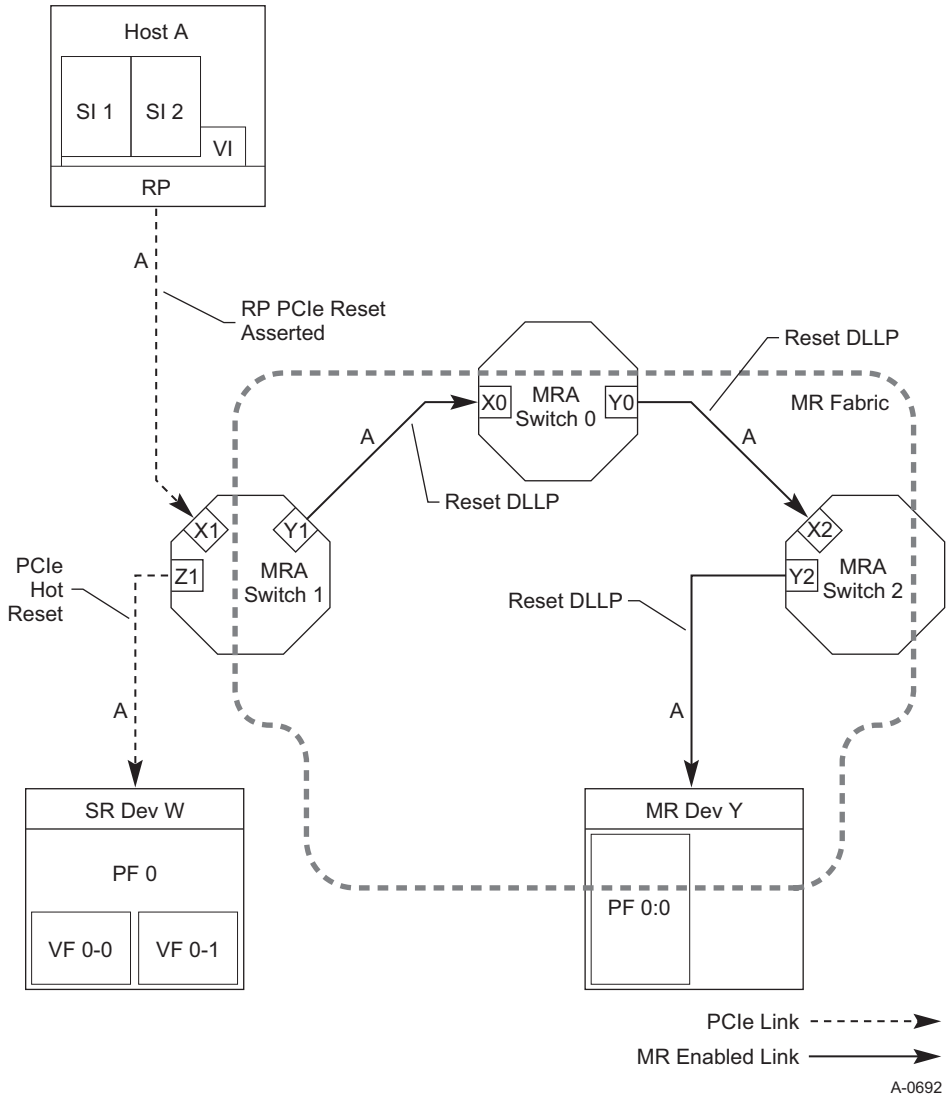

Figure 2-20: Reset DLLP Example: Host A's View

Table 2-4: Reset DLLP Example: Events and Actions

Component	Event	Action
Host A	Sends Hot Reset on RP_A to MRA Switch 1	
MRA Switch 1	Sees Hot Reset on Port X1 which is the Upstream Port of VH A	1.1 Discards all TLPs headed out Port X1 1.2 Sends Training Sequence 1 with Hot Reset asserted on Port X1 1.3 Sends Hot Reset out Port Z1 to Device W 1.4 Sends Reset DLLP Request for VH A out Port Y1 to MRA Switch 0 *Unlabeled Ports not affected as they are not part of VH A*
	All TLPs for VH A are discarded or marked for discard and Port X1 Retry Buffer empty	1.5 Nothing – VH A Upstream Port is not MR Link
	Sees Reset Ack DLLP on VH A from Port Y1	1.6 Knows MRA Switch 0 will not send any more TLPs for VH A through Port Y1 1.7 Stops resending Reset DLLP for VH A out Port Y1
MRA Switch 0	Sees Reset DLLP Request for VH A on Port X0 which is the Upstream Port of VH A	0.1 Starts discarding TLPs for VH A 0.2 Sends Reset DLLP Request for VH A out Port Y0 to MRA Switch 2 *Unlabeled Ports not affected*
	All TLPs for VH A are discarded or marked for discard and Port X0 Retry Buffer contains no TLPs for VH A	0.3 Send Reset Ack DLLP out Port X0 to MRA Switch 1
	Sees Reset Ack DLLP on VH A from Port Y0	0.4 Knows MRA Switch 2 will not send any more TLPs for VH A through Port Y0 0.5 Stops resending Reset DLLP for VH A out Port Y0
MRA Switch 2	Sees Reset DLLP Request for VH A on Port Z0 which is the Upstream Port of VH A	2.1 Starts discarding TLPs for VH A 2.2 Sends Reset DLLP Request for VH A out Port Y2 to Device Y *Unlabeled Ports not affected*
	All TLPs for VH A are discarded or marked for discard and Port X2 Retry Buffer contains no TLPs for VH A	2.3 Send Reset Ack DLLP out Port X0 to MRA Switch 0

Component	Event	Action
	Sees Reset Ack DLLP on VH A from Port Y2	2.4 Knows Device Y will not send any more TLPs for VH A through Port Y2 2.5 Stops resending Reset DLLP for VH A out Port Y2
Device W	Sees PCIe Hot Reset	W.1 Discards all TLPs, enters Reset W.2 Sends Training Sequence 1 with Hot Reset asserted
Device Y	Sees Reset DLLP Request on VH A	Y.1 Starts discarding TLPs for VH A
	All TLPs for VH A are discarded or marked for discard and no TLPs for VH A are in the Retry Buffer	Y.2 Send Reset Ack DLLP
Device X and Device Z	Nothing; not part of VH A	

2.3.2. RESET DLLP Format

Figure 2-21 shows the bit encoding of the Reset DLLP.

The A bit is 0 for a Reset Request (propagating downstream) and is 1 for a Reset Ack (propagating upstream).

The VH Group contains the upper bits of the VHN.

The Assert field contains one bit for each of 16 VHs within a VH Group. An Assert bit is 1 to indicate that the associated VH is in Reset and is 0 to indicate that the associated VH is not in Reset.

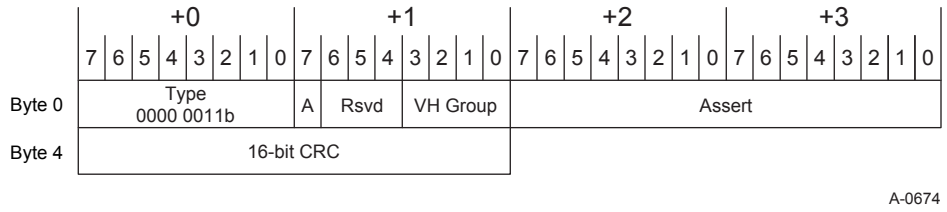

Figure 2-21: Reset DLLP

Reset DLLPs can be sent at any time a Link is in DL_Active. Reset Request DLLPs request a downstream Link partner enter or exit Reset state on one or more VHs. Reset Ack DLLPs indicate that the downstream Link partner has seen and processed the reset request.

Sending a Reset DLLP with the Assert bit set is making a promise about flushing of stale TLPs. In particular, a component sending a Reset Request or Ack DLLP with Assert == 1 is claiming that all TLPs from before the VH entered Reset have been flushed (either discarded or marked for later discard). This claim includes TLPs that are sitting in the Retry buffer. Consequently, typical implementations will delay sending the Reset DLLP until the affected TLPs are retired from the Retry Buffer using normal Ack protocol (TLPs transmitted during this time will be discarded at the remote end of the Link).

2.3.3. RESET DLLP Processing

The following sections describe Reset DLLP processing steps in an upstream and downstream ends of an MR Link. Recall that upstream and downstream are relative to a VH. In particular, one end of a Link may be using the upstream state machine for some VHs and the downstream state machine for other VHs.

The upstream and downstream state machines run in parallel on every VH.

2.3.3.1. Upstream State Machine

The following steps are used at the upstream end of a VH to enter Reset. This is triggered by a request to send a Reset DLLP. This request can occur for a variety of reasons (e.g., DL_DOWN, Reset DLLP Request, or Hot Reset on an upstream Switch Link, setting Secondary Bus Reset bit in some Type 1 Configuration header, etc.)

1. Reset requested.
2. Start discarding new TLPs received for this VH from this Link.
3. Start discarding new TLPs to be transmitted for this VH on this Link.
4. Discard or mark for discard any TLPs waiting to be sent for this VH on this Link.
5. Wait until the Retry Buffer contains no TLPs for this VH (i.e., they have been acknowledged and thus removed from the Retry Buffer).[7]
6. If this component advertised finite (VH VL) flow control credits, wait until all TLPs received for this VH from this Link have been discarded.
7. Schedule a Reset Request DLLP to be sent with this VH's Assert bit = 1.
8. Schedule resending the Reset Request DLLP approximately every 30 µs until a Reset Ack DLLP with the Assert bit = 1 is received (see Section 2.3.3.3 for details).
9. If a Reset Ack DLLP is received with this VH's Assert bit = 1, the remote end has entered Reset, stop scheduling Reset Request DLLPs for this VH (Reset Request DLLPs may continue to be sent if needed by other VHs).
10. When the remote end has entered Reset, a transmitter that was offered finite (VH VL) flow control credits by its Link Partner shall return any VL credits that were consumed by this VH.
    - If no TLPs have been transmitted on (VHx VLy) since the last time MRFC_INIT1_VH was entered, there are no VL credits to return.
    - If TLPs have been transmitted on (VHx VLy) then the transmitter shall adjust CREDIT_LIMIT for VLy as if it received an MRUpdateFC DLLP for (VHx VLy) for each credit type (see Section 2.4.1). This adjustment occurs using the following formula:

[7] This could be implemented by waiting until all TLPs, for any VH, have been flushed from the Retry Buffer that were in the Retry Buffer prior to step 3 (step 3 ensures that no new TLPs for this VH will enter the Retry Buffer).

- CREDIT_LIMIT_VL = (CREDIT_LIMIT_VL + CREDITS_CONSUMED_VHVL + INITIAL_CREDIT_LIMIT_VHVL – CREDIT_LIMIT_VHVL) mod 2^{Field_Size}
- Where CREDITS_CONSUMED_VHVL and CREDIT_LIMIT_VHVL are the values that existed prior to entry into MRFC_INIT1_VH and INITIAL_CREDIT_LIMIT is the flow control value recorded the last time MRFC_INIT2_VH was exited.

The following steps are used to exit Reset. This is triggered by the condition causing the entry into Reset going away (e.g., clearing Secondary Bus Reset, Physical LinkUp transitions from 0 to 1, etc.).

1. Schedule a Reset Request DLLP to be sent with this VH's Assert bit = 0.
2. Schedule resending the Reset Request DLLP approximately every 30 μs until Reset Ack DLLP with this VH's Assert bit = 0 is received (see Section 2.3.3.3 for details).
3. If a Reset Ack DLLP has been received with this VH's Assert bit = 0, the remote end has exited Reset, stop scheduling Reset Request DLLPs for this VH (Reset Request DLLPs may continue to be sent if needed by other VHs).

Reset Request DLLPs may be coalesced so that multiple scheduled events result in a single DLLP being transmitted.

Timely propagation of Reset is important. Components should send Reset Request DLLPs as soon as possible after starting to enter or exit Reset.

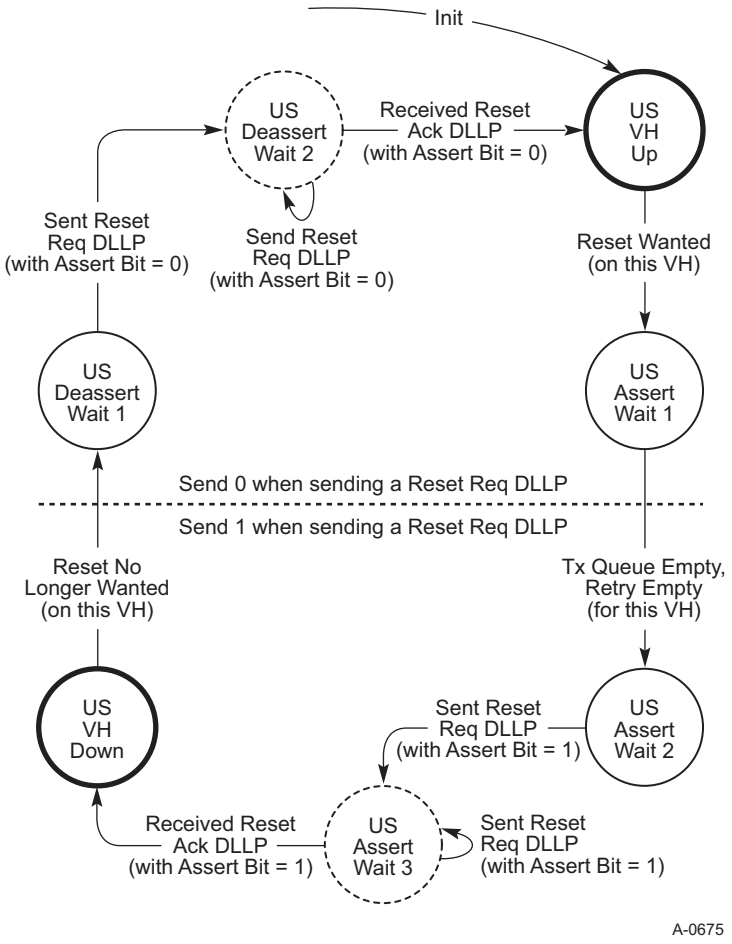

Figure 2-22: Upstream Link Partner RESET SM

2.3.3.2. Downstream State Machine

The following steps are used to enter Reset. This is triggered receiving a Reset DLLP on a Link with the Assert bit = 1.

1. Reset Request DLLP is received with this VH's Assert bit = 1.
2. Start discarding new TLPs received for this VH from this Link.
3. Start discarding new TLPs to be transmitted for this VH on this Link.
4. Discard or mark for discard any TLPs waiting to be sent for this VH on this Link.
5. A transmitter that was offered finite (VH VL) flow control credits by its Link Partner shall return any VL credits that were consumed by this VH.
    - If no TLPs have been transmitted on (VHx VLy) since the last time MRFC_INIT1_VH was entered, there are no VL credits to return.
    - If TLPs have been transmitted on (VHx VLy) then the transmitter shall adjust CREDIT_LIMIT for VLy as if it received an MRUpdateFC DLLP for (VHx VLy) for each credit type (see Section 2.4.1). This adjustment occurs using the following formula:

- CREDIT_LIMIT_VL = (CREDIT_LIMIT_VL + CREDITS_CONSUMED_VHVL + INITIAL_CREDIT_LIMIT_VHVL – CREDIT_LIMIT_VHVL) mod 2^{Field_Size}
- Where CREDITS_CONSUMED_VHVL and CREDIT_LIMIT_VHVL are the values that existed prior to entry into MRFC_INIT1_VH and INITIAL_CREDIT_LIMIT is the flow control value recorded the last time MRFC_INIT2_VH was exited.

6. All Functions in the VH enter the Reset.
7. Wait until the Retry Buffer contains no TLPs for this VH (i.e., they have been acknowledged and thus removed from the Retry Buffer).
8. If this component advertised finite (VH VL) flow control credits, wait until all TLPs received for this VH from this Link have been discarded.
9. Schedule a Reset Ack DLLP to be sent with the Assert bit = 1.
10. Schedule another Reset Ack DLLP to be sent whenever a Reset Request DLLP is received and the Assert bits in the Reset Request match what would be transmitted in the Reset Ack (i.e., retransmit the Ack in case it got lost).

The following steps are used to exit Reset. This is triggered by receiving a Reset DLLP on a Link with this VH's Assert bit = 0.

1. Reset Request DLLP is received with this VH's Assert bit = 0.
2. If all Functions are ready to exit Reset, Schedule a Reset Ack DLLP to be sent with this VH's Assert bit = 0.
3. Schedule another Reset Ack DLLP to be sent whenever a Reset Request DLLP is received.

Reset Ack DLLPs may be coalesced so that multiple schedule events result in one DLLP being transmitted.

Timely acknowledgement of Reset is important. Components shall respond with Reset Ack within 1.9 ms (+0%, -100%) and are strongly encouraged to respond much quicker. The 1.9 ms value is chosen to avoid inadvertent Link retraining caused by the Reset DLLP Forward Progress Timer (see Section 2.3.3.3).

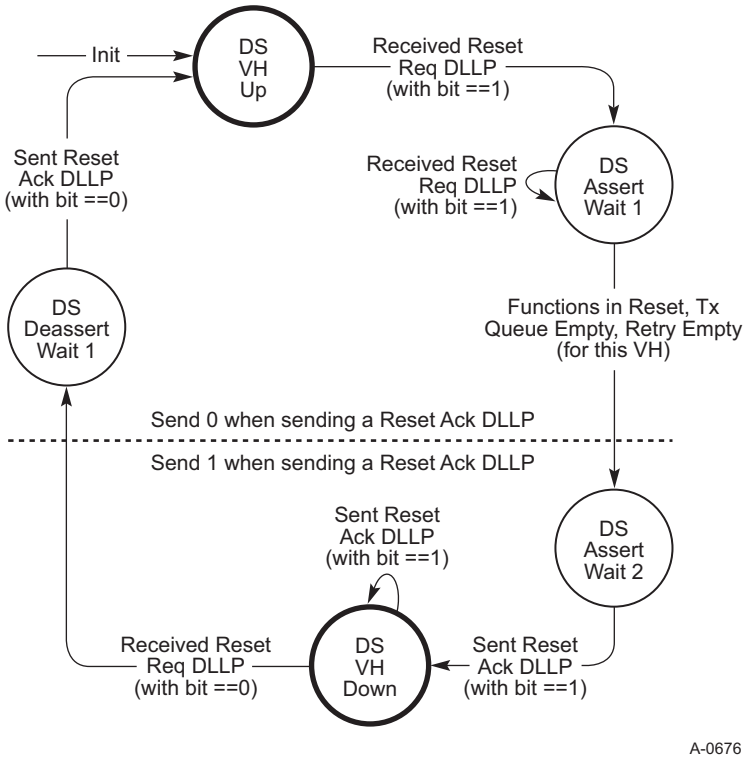

Figure 2-23: Downstream Link Partner RESET SM

2.3.3.3. Reset DLLP Reliability

A VH is considered to be waiting for Reset Ack when it is states US Assert Wait 3 or US Deassert Wait 2.

The Upstream component retransmits Reset Requests approximately every 30 μsec as controlled by the Reset Request Retransmit timer. This timer is enabled whenever the Link is in either the L0 or L0s state and any VH is waiting for Reset Ack. It resets and restarts when a Reset Request DLLP is transmitted for any VH group (either initial DLLP or a resend). It expires 30 μs after being started (+50%, -0%). When the timer expires, Reset Request DLLPs are scheduled to be sent for all VH groups that have some VH waiting for a Reset Ack.

The Upstream component also includes a Reset DLLP Forward Progress timer. This timer is enabled whenever the Link is in either the L0 or L0s state and any VH is waiting for Reset Ack. It resets and restarts when some VH enters any of the states US Assert Wait 1, US Assert Wait 2, US Assert 3, US Deassert Wait 1, or US Deassert Wait 2. It expires 2 ms after being started (+50%, -0%). When the timer expires, a Link retrain is requested.

The Upstream component also includes a 2-bit Reset Retrain counter. This counter is incremented when a Link retrain is requested due to the expiration of the Reset DLLP Forward Progress timer. This counter resets to 00b whenever the Reset DLLP Forward Progress timer is restarted. If this counter rolls over from 11b to 00b, the Link shall enter Detect by momentarily directing the LTSSM to the Disabled state.

Reset state machines are not affected by Link retraining (either initiated by the Reset DLLP Forward Progress Timer or through other means). DLLPs that were scheduled during the Link retrain shall be sent when the Link retrain completes.

Requests to retrain the Link initiated by this mechanism may be coalesced with requests to retrain the Link initiated by other mechanisms. For example, two "simultaneous" requests for a Link retrain due to (1) a REPLAY_NUM rollover and to (2) the Reset Forward Progress timer may result in either one or two retrain sequences.

2.3.3.4. Flow Control and Reset/DL_DOWN

In Multi-Root systems, Flow Control DLLPs can affect more than one VH. It is critical that a VH entering or exiting the Reset state does not disrupt other VHs. Flow Control credits on MR Enabled Links must be returned to the originator even when the VH that originated the discarded TLPs is in or is entering Reset.

For example, suppose TLPs enter a Switch on the MR Link associated with Port 1 destined for Port 2. If Port 2 enters Reset or DL_DOWN and thus discards these TLPs, credits must be returned in the appropriate VH of Port 1. This must occur whether or not Port 2 is an MR Link. If the associated VH of Port 1 is also in reset, credits are returned as a side effect of that reset.

Reset propagation may not be affected by Flow Control. Reset DLLPs must be sent independent of any Flow Control state.

Reset propagation is affected by the TLP Ack/Nak protocol. Allowing this avoids the complexity of editing the Retry Buffer to remove TLPs for VHs that are now in Reset.

2.3.3.5. Resets in Management VHs

In PCIe, Resets affect the entire component. In MR, only the indicated VH is Reset. If the VH being Reset is used for management of the MR Topology, other VHs are affected. Specifically:

- For Devices, a Reset in VH0 affects the entire component. An FLR directed to a BF also resets all Functions associated with that BF. An FLR to the Main BF Clears the MR Enable and NumVH fields and thus indirectly affects all Functions associated with all BFs.
- For Switches, a Reset in a Management VH, this affects that hierarchy and does not affect MR state. Furthermore, if the VS Suppress Reset Propagation bit is Set in the Management VS, resets seen at the upstream port of a VS do not propagate to downstream ports (see Section 4.3.5.2). This allows a more graceful failover to a backup MR-PCIM when the primary MR-PCIM fails (e.g., DL_Down at the MR-PCIM RP will not take down the entire MR topology).

2.4. MR Flow Control

2.4.1. FC Information Tracked by Transmitter

A transmitter shall track the following two quantities for each supported (VH VL) and VL. As in PCIe, Header and Data credits are tracked independently.

- CREDITS_CONSUMED
  - Computed and used in the same manner as in the PCIe protocol.
- CREDIT_LIMIT
  - Use in the same manner as in the PCIe protocol
  - Undefined at interface initialization
  - Set to the value indicated during MR Flow Control initialization
  - Updated based on MRUpdateFC DLLPs as described below

In MR, transmitting a TLP requires passing four gates instead of the two gates used in PCIe.

- One PH, NPH, or CPLH header VL credit is required.
- One PH, NPH, or CPLH header (VH VL) credit is required.
- Some number of PD, NPD, or CPLD data VL credits are required. The number of credits needed depends on the TLP size and uses the same rules as PCIe (and might be zero).
- Some number of PD, NPD, or CPLD data (VH VL) credits are required. The number of credits needed depends on the TLP size and uses the same rules as PCIe (and might be zero).

The transmitter gating function for a TLP is said to pass if the transmitter gating function passes for all four gates. If any gate fails, the transmitter must block transmission of the TLP. If CREDIT_LIMIT was specified as "infinite" during Flow Control initialization, then the corresponding gating function is unconditionally satisfied for that type of credit.

The Transmitter must follow the same ordering and deadlock avoidance rules as specified in the PCIe protocol. TLPs mapped to different VLs have no ordering relationship and must not block each other.

The transmitter gating function rules for (VH VL)s are the same as defined in the PCIe protocol.

The transmitter gating function rules for VLs are the same as defined in the PCIe protocol with the following exception:

- For Switches and Roots, a TLP heading upstream within its VH must also be gated unless, after sending the TLP, enough VL credits are available to send one additional maximum sized TLP of the same type.[8]

  This rule ensures that a TLP heading upstream cannot consume all available VL credits and thus cannot completely block TLPs heading downstream. [9]

TLPs associated with different VHs represent different flows and have no ordering relationship. TLPs blocked due to failure of the (VH VL) gating function should not block TLPs associated with other VHs mapped to the same VL.

---

[8] This rule avoids a deadlock that could have resulted from a single VL being used by multiple VHs in a mixed upstream / downstream configuration when combined with ACS redirection.

[9] This rule does not apply to Devices since Devices are always downstream in every VH.

If infinite credits were advertised for (VHx VLy) on some VHx and a given Credit Type, then the credit values must also be infinite for (VHz VLy) and that Credit Type on any other VHz.

The initial credit values for a Data Credit Type must match the corresponding Header Credit Type and vice versa. Specifically:

- For a given VL or (VH VL), if either the Posted Header or the Posted Data credit value is infinite, both values must be infinite.
- For a given VL or (VH VL), if the Non-Posted Header is infinite, the Non-Posted Data credit value must be infinite.
- For a given VL or (VH VL), if either the Completion Header or the Completion Data credit value is infinite, both values must be infinite.

If the credit values are infinite for VLy and (VHx VLy) for a given Credit Type, UpdateFC and MRUpdateFC DLLPs need not be sent for that Credit type. If they are sent, the credit value fields must be set to zero and must be ignored by the Receiver. The Receiver may optionally check for non-zero update values (in violation of this rule). If a component implementing this check determines a violation of this rule, the violation is a Flow Control Protocol Error (FPCE).

If the credit value for (VHx VLy) is non-infinite, then MRUpdateFC DLLPs are sent. MRUpdateFC DLLPs are ignored and do not update CREDIT_LIMIT values unless the Reset State Machine for that VH is in either the US VH Up or DS VH Up states (see Section 2.3.3)

- An MRUpdateFC DLLP with the value Update_Value causes the update the values of VL and (VH VL) CREDIT_LIMIT variables as follows:
  - Credits_Received = (Update_Value – CREDIT_LIMIT_VHVL) mod $2^{\text{Field Size}}$
  - CREDIT_LIMIT_VL = (CREDIT_LIMIT_VL + Credits_Received) mod $2^{\text{Field\_Size}}$
  - CREDIT_LIMIT_VHVL = Update_Value
  - Credits_Received is computed using the CREDIT_LIMIT_VHVL value *before* it is updated by this DLLP.
  - If CREDIT_LIMIT is infinite, it is not updated by this DLLP.

Otherwise, if credit value for (VHx VLy) is infinite, then PCIe Base UpdateFC DLLPs are sent. A transmitter may optionally raise a Receive Error if an MRUpdateFC DLLP with non-zero Update_Value is received.

- UpdateFC DLLPs update the value of VL CREDIT_LIMIT variable as follows:
  - CREDIT_LIMIT_VL = Update ValueFC

IMPLEMENTATION NOTE

VL Only Flow Control (Transmit)

As described in this section, MR Flow Control involves up to 4 transmit gates. Gates where infinite credits were advertised by the Link partner always succeed. If (VH VL) credits are infinite for some VL, TLP type, then UpdateFC DLLPs are used instead of MRUpdateFC DLLPs for that VL and

TLP type. For this case, some MRUpdateFC DLLPs are still sent during initialization (see sections 2.1.2.2 and 2.4.2)

A non-switch component can tell its Link Partner to advertise infinite (VH VL) credits by clearing the VH FC bit in the MRInit DLLP (see section 2.1.1). Components doing this do not have to implement the (VH VL) transmit gates since they will always be infinite and may implement only the VL transmit gates. Switches are not allowed to request this behavior, must set VH FC to 1b and thus must implement all 4 transmit gates.

2.4.2. FC Information Tracked by Receiver

A receiver shall track the following two quantities for each supported (VH VL) and VL.

- CREDITS_ALLOCATED
  - The initial value is Vendor Specific. This value was communicated to the transmitter using the MR Flow Control Initialization Protocol.
  - For (VH VL) credits, the minimum initial value credit values are as defined in the PCIe specification. For VL credits on Switch and Root Ports, the minimum initial credit values are doubled (e.g. for PH, 2 credits; for PD twice the largest possible setting of the Max_Payload_Size for the Port divided by FC Unit Size).
  - This value is incremented as processing (or discarding) of received TLPs makes additional receiver buffer space available.
  - Changes to this value are communicated to the transmitter using flow control update DLLPs.
  - If VL and (VH VL) credits are both finite, flow control update DLLPs may not be used to change the credit allocation for a VL or a (VH VL). All credits returned by Flow Control update DLLPs must be a result of processing or discarding of TLPs.
- CREDITS_RECEIVED
  - Computed in the same manner as in PCIe protocol.

If a Receiver advertises non-infinite (VH VL) credits, then it must send MRUpdateFC DLLPs whenever an FCP for that VL must be scheduled for transmission.

If a Receiver advertises infinite (VH VL) credits, then it must send UpdateFC DLLPs whenever an FCP for that VL must be scheduled for transmission.

When the Link is in the L0 or L0s Link State and after an MRUpdateFC has been sent four times with the same value, subsequent MRUpdateFC FCPs with the same value must be scheduled for transmission at least every 100 ms (-0%/+50%). Implementations should use this mechanism to reduce the amount of FCP traffic associated with inactive flows.

Independent of the above rule, when the Link is in the L0 or L0s Link state, some UpdateFC or MRUpdate FCP must be scheduled for transmission at least once every 30 µs (-0%/+50%), except when the Extended Sync bit of the Control Link register is set, in which case the limit is 120 µs (-0%/+50%).

- For non-infinite (VH VL) credits of types NPH, NPD, PH, and CPLH, an MRUpdateFC FCP for (VHx VLy) must be scheduled for Transmission each time either of these events occur:
  - No (VHx VLy) credits for a particular type are available and then one or more credits of that type are made available by processing of TLPs headed downstream within VHx.
  - For Switch and Root Ports, when exactly one (VHx VLy) credit for a particular type is available and then one or more credits of that type are made available by processing of TLPs headed upstream within VHx.
  - For non-infinite VLy credits, when no VLy credits for a particular type are available and then one or more credits of that type are made available by processing of TLPs headed downstream within VHx.
  - For Switch and Root Ports with non-infinite VLy credits, when exactly one VLy credit for a particular type is available and then one or more credits of that type are made available by processing of TLPs headed upstream within VHx.
- For non-infinite (VH VL) credits of types PD and CPLD, an MRUpdateFC FCP for (VHx VLy) must be scheduled for Transmission each time either of these events occur:
  - When the number of available (VHx VLy) credits is less than Max_Payload_Size and one or more units of that type are made available by processing of TLPs within VHx.
  - For Switch and Root Ports, when the number of available (VHx VLy) credits is less than 2*Max_Payload_Size and one or more units of that type are made available by processing of TLPs within VHx.
  - For non-infinite VLy credits, when the number of available VLy credits is less than Max_Payload_Size and one or more units of that type are made available by processing of TLPs within VHx.
  - For Switch and Root Ports with non-infinite VLy credits, when the number of available VLy credits is less than 2*Max_Payload_Size and one or more units of that type are made available by processing of TLPs within VHx.
  - For a multi-Function device, largest Max_Payload_Size setting across all Functions in all VHs must be used. For an MR Switch, the largest Max_Payload_Size setting across all VS Bridges mapped to this Port in any VH must be used.
- For infinite (VH VL), credits UpdateFC FCPs for (VLy) must be scheduled for Transmission as specified in the *PCI Express Base Specification*.
- A Port which has advertised infinite VH credits on (VHx VLy) must schedule periodic MRUpdateFC DLLPs with a Header or Data Credit Count of zero for (VHx VLy), with any legal TT and Credit Type, at rate of at least one every 1 ms (-0%/+50%). Sending of these periodic MRUpdateFC DLLPs is discontinued when a TLP or MRUpdateFC is received on (VHx VLy). It is recommended that implementations sending MRUpdateFC DLLPs due to this requirement do so infrequently so as to not adversely affect link availability for other DLLPs and TLPs. This requirement is addition to that of sending UpdateFC for VLy, as previously described in this section.
- MRUpdateFC FCPs may be scheduled for Transmission more frequently than is required.

IMPLEMENTATION NOTE

VL Only Flow Control (Receive)

Components must respond to the VH FC bit in the MRInit DLLPs. If the received VH FC bit is 0b, they must offer infinite (VH VL) credits. In this situation, the receiver depends on VL credits to prevent input buffer overflow. If the received VH FC bit is 1b, the Link partner is not placing a requirement on the receiving component so it may offer either infinite or non-infinite (VH VL) credits.

Components that advertise infinite (VH VL) credits send flow control updates using UpdateFC DLLPs instead of MRUpdateFC DLLPs. This is true whether they advertise infinite credits on their own of because they were commanded to by their Link Partner's VH FC bit. As described above, during initialization, some MRUpdateFC DLLPs are sent. These DLLPs will force the exit from state MRFC_INIT2_VH if some MRInitFC2_VH DLLPs were lost.

Infinite (VH VL) does not alter Flow Control Initialization (states MRFC_INIT1_VH, MRFC_INIT2_VH are still involved). Doing so avoids a more complicated initialization sequence and allows the Flow Control Negotiation pending bit to remain meaningful.

A component that offers infinite (VH VL) credits and sends MRInit DLLPs with the VH FC bit clear is a VL Flow Control Only Component. Neither its transmitter nor its receiver need to track (VH VL) credits and it can mostly ignore MRUpdateFC DLLPs.

2.4.3. Electrical Idle Inference

MR Links use MRUpdateFC DLLPs instead of or in addition to UpdateFCs. For such links, the Electrical Idle inference condition for state L0 shall be sensitive to either style of Flow Control update DLLP. The conditions where Electrical Idle can be inferred for other states are not affected.

In Section 4.2.4.3, Table 4-6 of the *PCI Express Base Specification 2.0*, the first row is modified to read:

Table 2-5: Modified Electrical Idle Inference Conditions

State	2.5 GT/s	5.0 GT/s
L0	Absence of any Flow Control Update DLLPs (either MRUpdateFC or UpdateFC) or alternatively a SKP Ordered Set in 128 µs window	Absence of any Flow Control Update DLLPs (either MRUpdateFC or UpdateFC) or alternatively a SKP Ordered Set in 128 µs window

Similarly, the first Note in Section 4.2.6.5 of the *PCI Express Base Specification 2.0* is modified to read:

- "Note: As described in Section 4.2.4.3, an Electrical Idle condition may be inferred on all Lanes under any one of the following conditions: (i) absence of Flow Control Update DLLPs (either UpdateFC or MRUpdateFC) in any 128 µs window, (ii) absence of a SKP Ordered Set in any of the configured Lanes in any 128 µs window, or (iii) absence of Flow Control Update DLLPs (either UpdateFC or MRUpdateFC) or a SKP Ordered Set in any of the configured Lanes in any 128 µs window."

2.5. MR Message Processing

2.5.1. Interrupts

Interrupt processing occurs within a VH. The associated TLPs contain a TLP Prefix allowing all components to route them appropriately.

MSI and MSI-X Interrupts are indistinguishable from other Memory Write TLPs.

INTx Interrupts are represented using ASSERT_INTx/DEASSERT_INTx messages per the *PCI Express Base Specification*. In MR, these messages are queued and ordered within a VH.

2.5.1.1. INTx Device Processing

In PCIe, INTx messages are emitted by Devices. An ASSERT_INTx message is emitted when an interrupt condition is signaled. A DEASSERT_INTx message is emitted when the interrupt condition has been satisfied.

In MR, INTx messages are emitted within a VH. If a function within a VH signals or satisfies an interrupt, ASSERT_INTx/DEASSERT_INTx messages are emitted within that VH. These messages are unrelated to INTx messages issued in any other VH.

2.5.1.2. INTx Switch Processing

In PCIe, INTx messages are processed by Switches. Each downstream Switch Port tracks an internal INTx wire for each of INTA/B/C/D. These INTx wires are combined into four INTx wires at the upstream Switch Port. Transitions of the combined INTx wires trigger sending of INTx messages out the upstream Switch Port.

Similarly, in MR, INTx messages are processed by Virtual Switches. Each virtual downstream Switch Port tracks an internal INTx wire for each of INTA/B/C/D. The INTx wires are combined into four INTx wires at the virtual upstream Switch Port. Transitions of the combined INTx wires trigger sending of INTx messages out the virtual upstream Switch Port.

Switch reconfiguration will also affect INTx. For example, when a Virtual Device is unmapped from a Virtual Switch, the virtual downstream Switch Port sees DL_Down. This virtual downstream Switch Port follows PCIe rules by deasserting the internal INTx wires. If this results in a transition of the combined INTx wires, a DEASSERT_INTx message is sent out the virtual upstream Switch Port.

2.5.1.3. INTx Root Port Processing

In PCIe, the Host Processor tracks the INTx assert/deassert state. If INTx is asserted, and software has not masked processing, an interrupt is sent to host processor.

In MR with a PCIe Root Port, this is unchanged. By the time the INTx message is seen by the Root Port, the TLP Prefix has been dropped and the message is indistinguishable from non-MR usage.

In MR within an MR Root Port, the INTx messages are tagged with a VH and the RP must track independent INTx assert/deassert state for each VH.

2.5.2. PME Turn Off Processing

PCIe Switches must perform a PME Turn Off scoreboard function. MR Switches must do so as well within each VS. See Section 7.7 for details.

2.5.3. PM_PME Processing

MR Switches must convert PM_PME messages into Beacon/WAKE# indications in certain situations. See Section 7.6.1 for details.

2.6. Miscellaneous Changes

There are certain secondary effects caused by other MR Protocol changes.

- TLPs in a retry buffer are slightly bigger since it includes a TLP Prefix. This has a minor ripple effect (e.g., Ack/Nak timer values may need minor adjustments).
- The TLP Prefix is queued with the TLP. This increases the amount of buffering required for the TLP header.

2.7. Miscellaneous Non-Changes

The following items are not affected by Multi-Root operations.

- Locked transactions are processed by MR Switches within each VS. Locked transactions in one VS will not affect another VS.
- Port arbitration occurs within each VS. Software in a VH may control Port arbitration using the optional VC Extended Capability in each bridge of the VS.
- Function arbitration occurs within each VH. Software in a VH may control Function arbitration using the optional MFVC Extended Capability in Function 0 of each VH.
- Values reported by the Power Budgeting Capability reflect the entire component. No attempt is made to split reported power consumption between VHs.

3

3. Initialization and Resource Allocation

3.1. MR Topology Initialization

MR Fabrics consist of a collection of the following components interconnected in a mesh topology:

- One or more MRA Switches
- One or more PCIM Capable Switch Management Ports
- Zero or more non-PCIM Capable RPs
- Zero or more Devices – MRA, SR, PCIe Device[10]

Unlike conventional PCI Express, MR topologies need not be a tree. MR Topologies can be an arbitrary mesh that may contain loops. Each VH sees a tree structure consisting of a subset of the overall MR Topology.

Because MR Topologies are not a tree, they have no single notion of Link upstream and downstream directions. Links have a Physical Link Direction (used in the Physical Layer Link training process) and a Physical Port Direction (used to determine how and where link control happens). In addition, each VH using a Link has a Logical direction which may differ from the Link's Physical directions.

There are a number of steps or phases used to initialize and configure the above components into an MR Topology. These steps are:

1. Initial State after Reset
2. PCIM Location Policy Decision
3. Topology Discovery
4. Component Discovery
5. Mapping Policy Decision
6. Mapping Implementation
7. Virtual Hierarchy Enumeration

Each step is described in more detail below.

[10] Useful MR Topologies will eventually have at least one device, but this is not strictly required. In particular, no devices might initially be present with devices being added later using Hot-Add operations.

3.1.1. Initial State After Fundamental Reset

MRA Switch Ports are configured into one of two Initial Port Types: PCIM Capable Switch Port and Non-PCIM Capable Switch Port. This configuration is set after Fundamental Reset using a Vendor Specific mechanism. The configuration mechanism can change Port type assignments, but the results of such a change are not visible until the next Fundamental Reset.

The initial configuration of every MR Switch must have at least one Switch Management Port. This Port can be either a PCIe PCIM Capable Switch Port or a Vendor Specific non-PCIe Switch Management Port.

The initial configuration of every MR Switch may have any number of Non-PCIM Ports (including zero).

Initial Port Type is indicated by the setting of a few key characteristics of a Port. These characteristics are configurable so that MR-PCIM software can change PCIe Port behavior from the initial settings. Port Type configurability for non-PCIe Switch Management Ports is Vendor Specific.

IMPLEMENTATION NOTE

MR Switch Used as a PCIe Switch

Vendor specific settings could be used to configure an MR Switch so it operates as a PCIe Switch. The PCIe Upstream Port could be a PCIM Capable Switch Port, allowing it to be used for reconfiguration. PCIe Downstream Ports would be Non-PCIM Ports with their Port_Direction set as downstream and their Port VH 0 mapped to some downstream P2P Bridge of the VS associated with the upstream Port.

3.1.1.1. PCIM Capable Switch Ports

A PCIM Capable Switch Port i is defined as a Switch Port with the following characteristics:

- Port[i].Port_Direction indicates an Upstream Port.
- Some VS[j] has mapped {Port[i], Port VH 0} to its Upstream Bridge.
- The upstream P2P Bridge Configuration header for VH 0, VS[j] has a full MR-IOV Capability.
- The bit j of the VS Authorization Bitmap is Set.
- The Management VS value is Vendor Specific. It could be j or it could be some other VS.
- VS[j] contains sufficient Enabled Downstream P2P Bridges to manage the system. Specifically, VS[j] must contain a Downstream P2P Bridge for every Port that MR-PCIM needs to be simultaneously mapped into the VH. For an *n*-Port MR Switch, this is at most *n-1* Downstream P2P Bridges and can be reduced based on system configuration (e.g., Ports that can only connect to PCIe RPs do not need a Downstream P2P Bridge).

These settings ensure that MR-PCIM could manage the Switch using this Port. MR-PCIM is not required to be present on this Port. The RP running MR-PCIM could be directly attached to this Port or could be connected indirectly using one or more MR or PCIe Switches.

This Port is authorized. In particular, any software using it will be allowed to manage the Switch. Software can later de-authorize the Port if desired.

Note: Since a VS has a single upstream bridge, these rules imply that every Potential PCIM Port will be associated with a distinct VS.

IMPLEMENTATION NOTE

Management Port Characteristics

PCIM Capable Switch Ports need not be full width "expensive" Ports. They could be x1 Ports intended just to support management.

PCIM Capable Switch Ports may or may not be enabled as MR Links. The Link MR-IOV Enable bit default value is Vendor Specific and could be Set using a Vendor Specific mechanism (see Section 4.3.3.2).

3.1.1.2. Non-PCIM Capable Switch Ports

A Non-PCIM Capable Switch Port j is defined as a Switch Port with the following characteristics:

- Port[j].Port_Direction is Vendor Specific.
- No Authorized VS has mapped {Port[j], any Port VH} to its upstream bridge.
- Mapping of Port[j] into bridges in VSs is otherwise Vendor Specific.
  - Any Port VH of Port[j] may be mapped to any downstream bridge of any VS.
  - Any Port VH of Port[j] may be mapped to the upstream bridge of any non-authorized VS. This upstream bridge may or may not have an MR-IOV capability in its Type 1 Configuration header. If it has one, only selected fields in the Type 1 header are operational and none of the MR-IOV tables located in memory are visible. See Section 4.3 for details.
  - Port VHs of Port[j] need not be mapped into any VS.

These settings ensure that Initial MR-PCIM will never be present on this Port. This Port can be directly or indirectly via Switches connected to Devices, Root Ports, or Bridges.

This Port is not authorized. Attempts by software attached to this Port to configure the MR Switch will fail unless the Port is later authorized.

3.1.1.3. Non-PCIe Switch Management Ports

A non-PCIe Port may also be used to manage MR Topologies. These Ports consist of Vendor Specific hardware that appears to an MR Switch as an upstream Port. Such Ports have a Port Table entry with the Non-PCIe bit set.

This Vendor Specific hardware allows MR-PCIM to issue and respond to the subset of PCIe transactions needed to manage the MR Topology.

- Such hardware must allow MR-PCIM to issue:
  - Configuration Read and Write Requests (Type 0 and Type 1) of sizes 1, 2, and 4 bytes, naturally aligned.
  - Memory Space Read and Write Requests (32-bit and 64-bit addressing) of sizes 1, 2, 4, and 8 bytes, naturally aligned.
  - Message Transactions normally issued by Root Ports (e.g., PME Turn Off messages).
- Such hardware must allow MR-PCIM to respond to:
  - Completions related to the above (including completion status and support for Completion Timeout)
  - Posted Memory Write transactions for MSI Interrupts
  - Message Transactions (e.g., INTx, PME_TO_Ack, PM_PME, Errors, …)

This is the minimum support needed to configure the MR Topology. Additional support may be required if MR-PCIM uses the MR Topology for general I/O (e.g., Logging). Additional support may also be required if Vendor Specific device management software also needs to use this Port.

A Non-PCIe Switch Management Port i is defined as a Port with the following characteristics:

- Port[i].Port_Direction indicates an Upstream Port.
- The Port operates in Base PCIe mode (the Link cannot be MR Enabled).
- Some VS[j] has mapped {Port[i], Port VH 0} to its upstream bridge. This mapping may be fixed.
- The upstream P2P Bridge Configuration header for VH 0, VS[j] has a full MR-IOV Capability.
- The bit j of the VS Authorization Bitmap is Set. This bit may be read only.
- The Management VS value is Vendor Specific. It could be j or it could be some other VS.
- VS[j] contains enough Enabled Downstream P2P Bridges to access all Ports of the MR Switch.

These settings ensure that MR-PCIM could manage the Switch using this Port. MR-PCIM is not required to be present on this Port.

This Port is authorized. In particular, any software using it will be allowed to manage the Switch.

Note: Since a VS has a single upstream bridge, these rules imply that every Non-PCIe Switch Management Port will be associated with a distinct VS.

3.1.1.4. Initial State Example

Figure 3-1 shows an example MR Topology. Both Switches are MRA Switches. Root Ports RP 0, RP 1, and RP 2 are capable of running MR-PCIM software and configuring the topology. RP 3 is not permitted to run MR-PCIM. Devices are a mixture of Base PCIe, SR Aware, and MR Aware Devices.

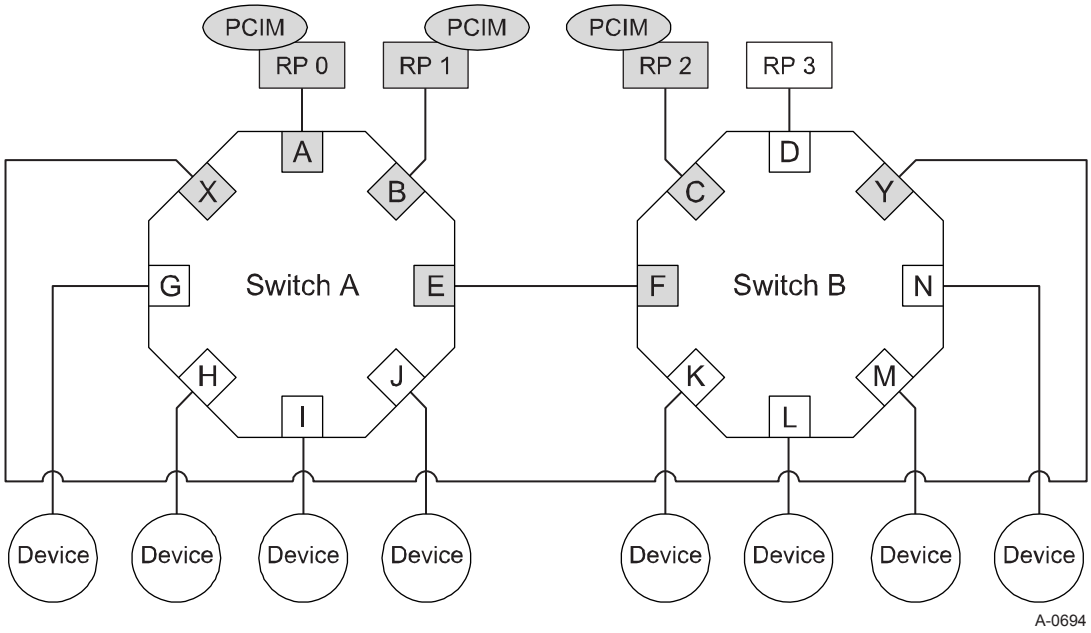

Figure 3-1: Example MR Topology

In this example, assignments are show in Table 3-1.

Table 3-1: Port Types – Example MR Topology

Port	Initial Port Type	Reason
A, B, C	PCIM Capable Switch Port	After Reset, RP 0, RP 1, or RP 2 are allowed to run MR-PCIM.
D	Non-PCIM Capable Switch Port	After Reset, RP 3 is not allowed to run MR-PCIM.
E, X	PCIM Capable Switch Port	If RP 2 runs MR-PCIM, this setting allows it to manage Switch A after Reset.
F, Y	PCIM Capable Switch Port	If either RP 0 or RP 1 runs MR-PCIM, this setting allows it to manage Switch B after Reset.
G, H, I, J	Non-PCIM Capable Switch Port	Devices never run MR-PCIM.
K, L, M, N	Non-PCIM Capable Switch Port	Devices never run MR-PCIM.

3.1.2. Initial MR-PCIM Location Policy

After Fundamental Reset, a Vendor Specific mechanism selects the Initial MR-PCIM location.

Only the selected system is allowed to access the MR topology. A Vendor Specific mechanism prevents other RPs and Non-PCIe Switch Management Ports from accessing the MR topology.

The Initial MR-PCIM software must be able to manage the entire MR Topology. To do so, it must be connected to a PCIM Capable Switch Port on every MR Switch. This connection can be direct or indirect using additional Switches.

3.1.2.1. Initial MR-PCIM Location Example

Continuing with the example from Section 3.1.1.4, assume that RP 0 is chosen for the Initial MR-PCIM. Vendor Specific mechanisms are used to prevent RP 1 and RP 2 from accessing the MR Topology; e.g., the affected processors might be powered off or held in reset.

3.1.3. Topology Discovery

The selected Initial MR-PCIM can connect to the first MR Switch in a variety of ways:

- Base PCIe RP directly connected to a PCIM Capable Switch Port on the MR Switch. The Link trains in Base PCIe mode. Only VH 0 of this Port will be used.
- MR enabled RP directly connected to a PCIM Capable Switch Port on the MR Switch. The Link trains in MR Enabled mode. VH 0 of this Port will be used by MR-PCIM to manage this Switch.
- Vendor Specific non-PCIe interface connected to a Non-PCIe Switch Management Port on the MR Switch.

In each of these cases, the upstream P2P Bridge first seen by MR-PCIM contains the MR-IOV capability and is mapped to VH 0 of an authorized VS. MR-PCIM uses this capability to configure the Switch.

MR-PCIM invents a unique number for the Switch and writes it to the MR Switch Number field. This number by used by MR-PCIM to detect topology loops during the enumeration process.

MR-PCIM then uses the Port table to enable additional links and to manage their Link_Direction. MR-PCIM might use Vendor Specific knowledge of the topology to aid in this process if, for example, a system is using Link_Direction to prevent Link training and thus hold off some RPs from seeing the hierarchy until it is configured.

For each Port that trains as a downstream Port, MR-PCIM will examine the Link Partner Training Status fields in the Port Table. For each Link Partner that is an Authorized Port on an MR Switch, MR-PCIM will determine whether the Port is mapped into the MR-PCIM VS and, if necessary, map the Port into an unused downstream bridge. It will then establish a Bus Number range for the downstream bridge that allows PCIe Configuration transactions to be issued to the Link Partner.

MR-PCIM then probes the Configuration header of the MR Switch attached to the Port.

- If the component is a "new" MR Switch because the Switch's MR Switch Number field has a number not assigned by MR-PCIM in this enumeration cycle, the enumeration process repeats to configure the new Switch.
- If the component is an "old" MR Switch that MR-PCIM has seen before, the MR Switch Number and connection information is noted but further enumeration is not needed via this connection. Note that enumeration of the "old" Switch may not be complete, but it will be completed using other links into the Switch.

3.1.3.1. MR-PCIM Topology Discovery Example

Figure 3-2 expands on the example shown in Figure 3-1 adding possible initial Link directions.

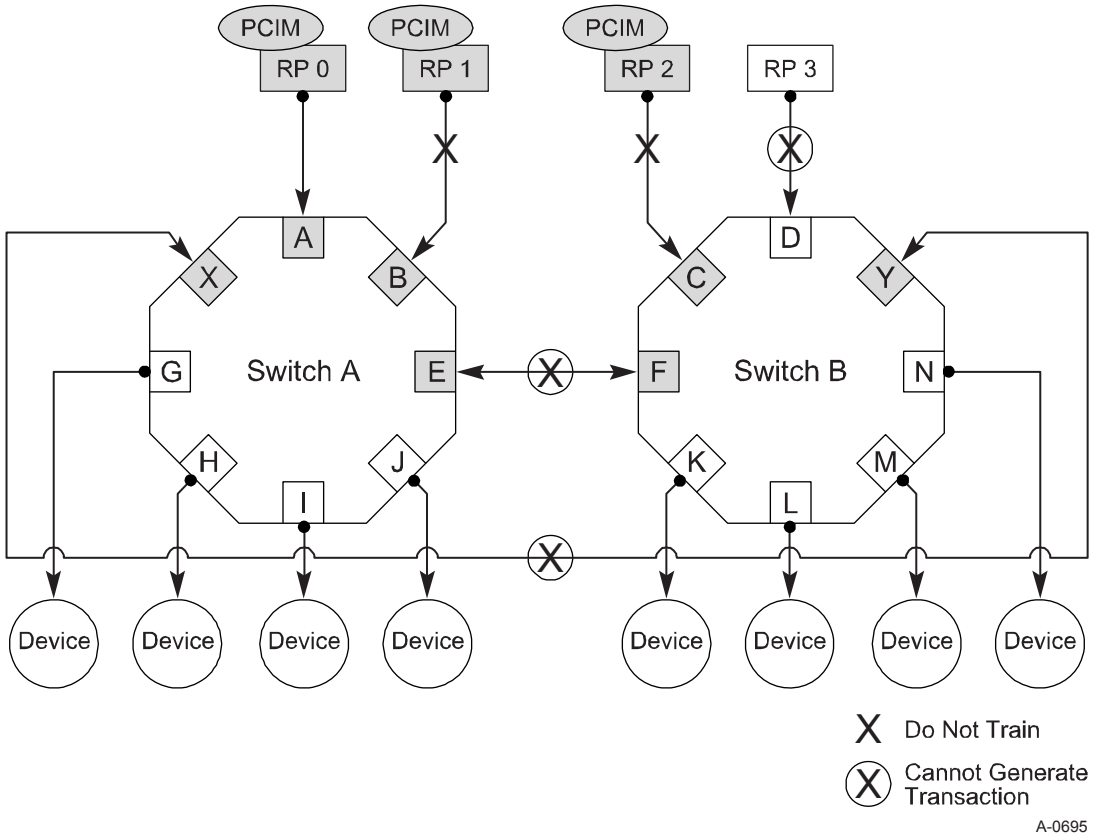

Figure 3-2: Example MR Topology with Initial Link Directions

The Links between Ports E and F and between Ports X and Y do not train since each Link consists of two upstream Ports. Since RP 0 was chosen as the location of the Initial MR-PCIM, the Link between RP 0 and Switch A trains. Vendor Specific mechanisms are used to prevent RP 1 and RP 2 from starting (links may train but no transactions will be generated from these RPs). The Link between RP 3 and Port D cannot train since it consists of two downstream Ports.

MR-PCIM software operating in RP 0 could proceed as follows:

1. Reads Type 1 Configuration header at Port A and detects MR-IOV capability indicating an MR Switch.
2. Configures BAR registers in Switch A so that the Port Table can be examined.
3. Assigns MR Switch Number 42 to Switch A.
4. Detects that Port A, VH 0 was mapped to VS n.
5. Notices Link attached to Port E detected something present but did not train. Switches Link_Direction for Link E to downstream thus allowing the E to F Link to train.
6. Notices Link attached to Port X detected something present but did not train. Switches Link_Direction for Link X to downstream thus allowing the X to Y Link to train.
7. Notices that Links G, H, I, and J trained as downstream. Each Port's Link Partner Training Status indicates no MR Switch is connected so no additional enumeration is needed at this time.
8. Notices that Port E is connected to an Authorized MR Switch. If needed, maps Port E to some downstream bridge of VS n of Switch A (it might already be mapped). Using this downstream bridge, Switch B is enumerated.
9. Reads the Type 1 Configuration header of Port F and detects MR-IOV capability indicating an MR Switch.
10. Configures BAR registers in Switch B so that the Port Table can be examined.
11. Assigns MR Switch Number 86 to Switch B.
12. Detects that Port F, VH 0 was mapped to VS m.
13. Notices that Links K, L, M, and N trained as downstream. Each Port's Link Partner Training Status indicates no MR Switch is connected so no additional enumeration is needed at this time.
14. Notices Link attached to Port D detected something present but did not train. Switches Link_Direction for Link D to upstream thus allowing the D to RP 3 Link to train. This step might be delayed until later if preventing Link training was the Vendor Specific mechanism used to hold off RP 3 from enumerating the topology.
15. Notices that Port X is connected to an Authorized MR Switch. If needed, maps Port X to some downstream bridge of VS n of Switch A (it might already be mapped). Using this downstream bridge, Switch B is enumerated. It then reads the Type 1 Configuration header of Port Y, and detects MR-IOV capability indicating an MR Switch. The previously assigned MR Switch Number of 86 indicates that Port X is a second path to Switch B. Since Switch B has been (or is still being) enumerated using Port E, no further enumeration is needed using Port X.

This is an example; other enumeration orderings are also valid. Note that the Link directions of the E to F and X to Y links depend on the chosen ordering. If, for example, instead of changing the Link_Direction of Port X in Step 6, the Link_Direction of Port Y were changed later during the enumeration of Switch B, the X to Y Link would train in the opposite direction.

Note: MR-PCIM assigns MR Switch registers locations in PCIe Memory Space during this discovery process. These assignments may change in subsequent software.

3.1.4. Component Discovery

Topology Discovery allows MR-PCIM to know what MR Switches exist and how they are interconnected. Component Discovery is now used to locate MR Devices and non-MR Components.

This process examines the Config Space of every Component. This information is gathered to drive Switch and Device Configuration Policy Decisions.

Information gathered from MR Devices includes:

- MaxVH
- Function number(s) of all BFs
- VF Mapping Supported/VF Migration Supported/VF MVF Region

Information gathered from MR Switches includes:

- MaxVH for each Port
- Number of VSs/Number of Bridges for each VS

Information gathered from every Link includes:

- Link Width and Speed

Information may not be available for Root Ports (MR or PCIe). The Vendor Specific mechanisms used to hold off transactions might also prevent the Link from training so the Link Partner Training Status may not yet be meaningful.

3.1.4.1. Component Discovery Example

Figure 3-3 expands on the example shown in Figure 3-2 adding Device characteristics.

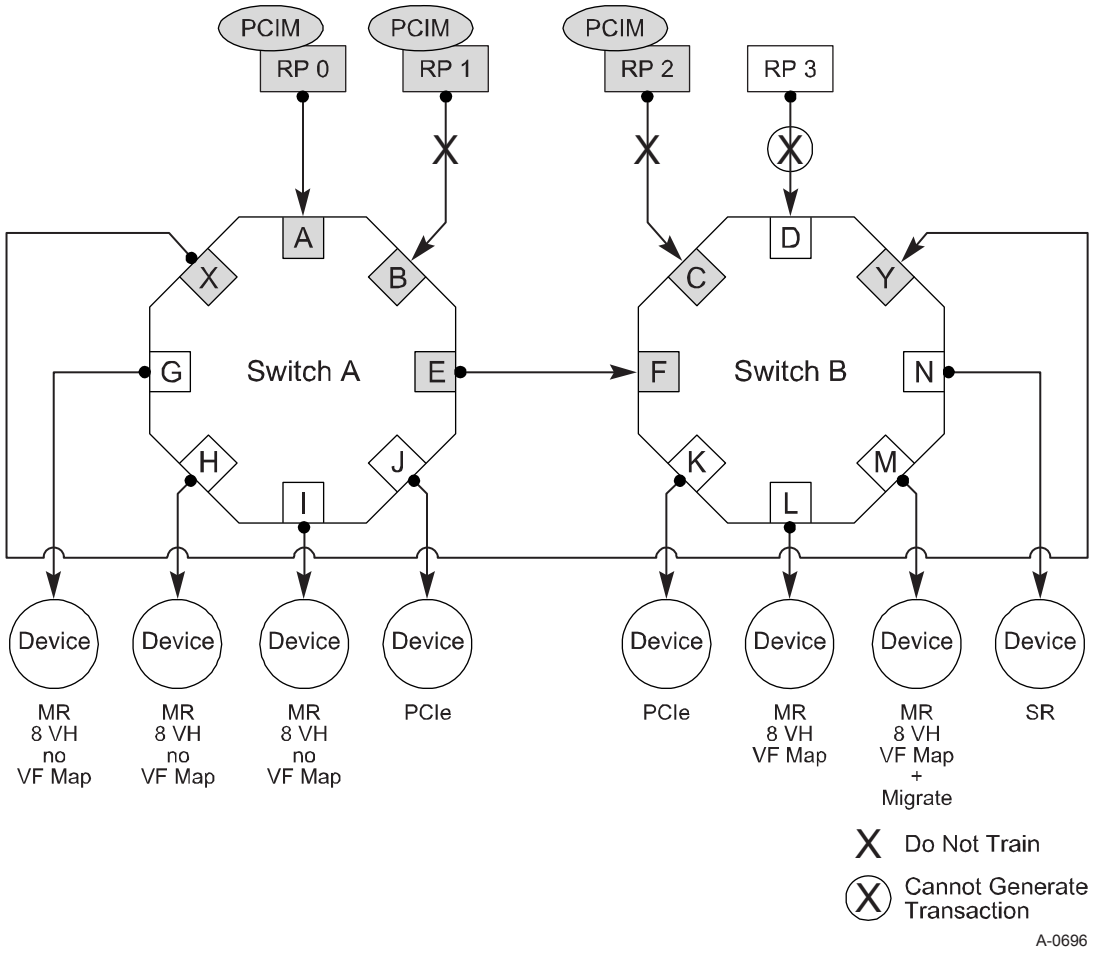

Figure 3-3: Example MR Topology with Component Discovery Details

This example assumes:

- Single Function MR Devices attached to Ports G, H, and I:
  - Each Device supports 8 VHs.
  - None of the Devices support VF Mapping or VF Migration.
- Base PCIe Devices attached to Ports J and K
- Single-Function MR Device attached to Port L:
  - The Device supports 8 VHs.
  - VF Mapping is supported with 32 LVFs and 32 MVFs.
  - VF Migration is not supported.

- Single Function MR Device attached to Port M:
  - The Device supports 8 VHs.
  - Both VF Mapping and VF Migration are supported with 32 LVFs and 30 MVFs.
  - VF and Function MVF Regions fully overlap.
- Single Root Device attached to Port N

3.1.5. VH and VF Mapping Policy

Out-of-Scope mechanisms are used to decide what portions of the MR Topology should be assigned to each VH.

3.1.5.1. Example VH and VF Mapping Policy

Table 3-2 expands on the example shown in Figure 3-3 adding VF and VF Mapping Policy Decision information.

Table 3-2: Example MR Topology VH and VF Mapping Policy

VH	Authorized	VH Mapping	VF Mapping
RP 0	Yes (MR-PCIM)	VS in Switch A and B Uses E to F Inter-Switch Link VH in Devices G, H, I, L, and M	Device L: 16 VFs Device M: 8 VFs (2 unpopulated)
RP 1	No	VS in Switch A VH in Devices G and H Device J	
RP 2	Yes (Backup MR-PCIM)	VS in Switch A and B Uses F to E Inter-Switch Link VH in Devices G, H, I, L, and M Device K	Device L: 8 VFs Device M: 8 VFs (2 unpopulated)
RP 3	No	VS in Switch A and B Uses Y to X Inter-Switch Link VH in Devices G, H, L, and M Two VHs in Device I Device N	Device L: 8 VFs Device M: 8 VFs (2 unpopulated)

3.1.6. VH and VF Mapping Implementation

Once the assignment policy is decided, MR-PCIM configures the Components to implement it. This involves writing the various tables in MR Switches and Devices. These tables are described in Sections 4.2 and 0.

All configuration activity occurs in the MR-PCIM VH. The order software uses for this initial configuration writes is unspecified. Configuration and memory requests originate exclusively from MR-PCIM.

3.1.6.1. Example Topology: Switch Implementation

The view from each Root Port is shown in Figure 3-4 through Figure 3-7. Switch VS Table Programming to implement this is shown in Table 3-3 and Table 3-4.

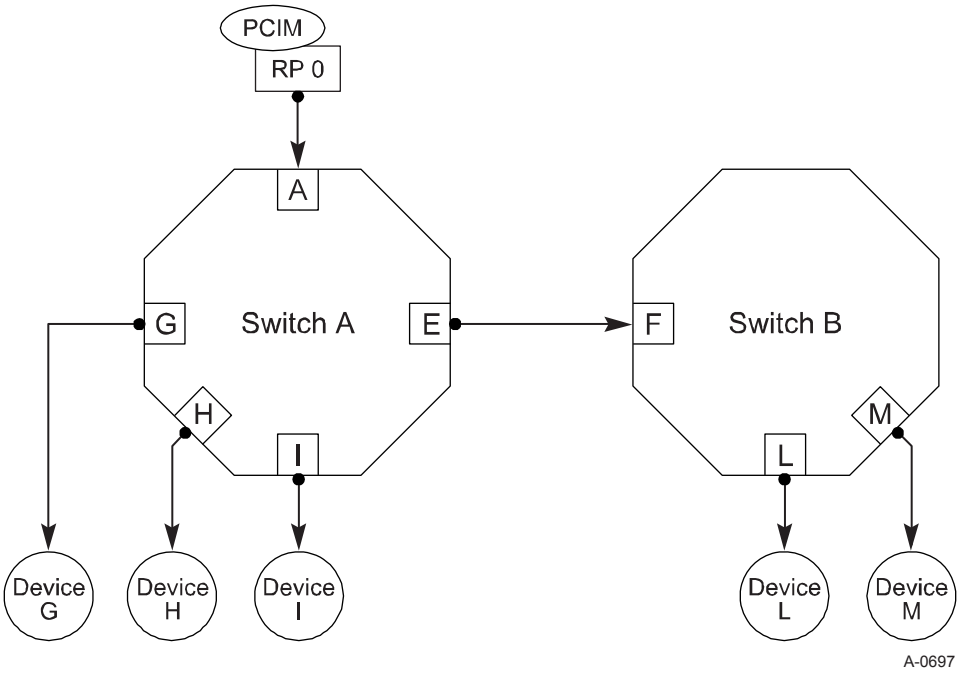

Figure 3-4: Example MR Topology: RP 0 View

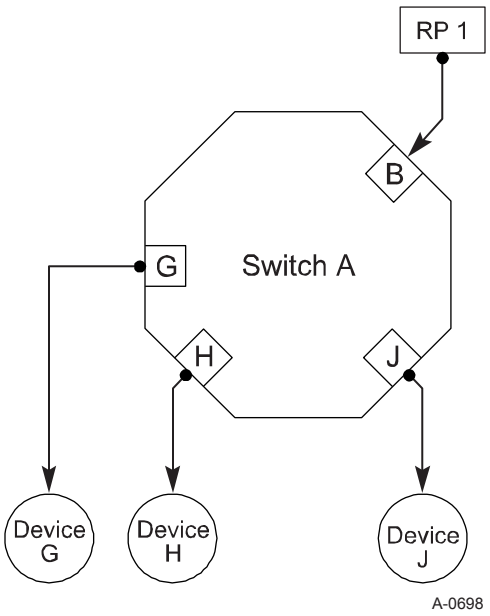

Figure 3-5: Example MR Topology: RP 1 View

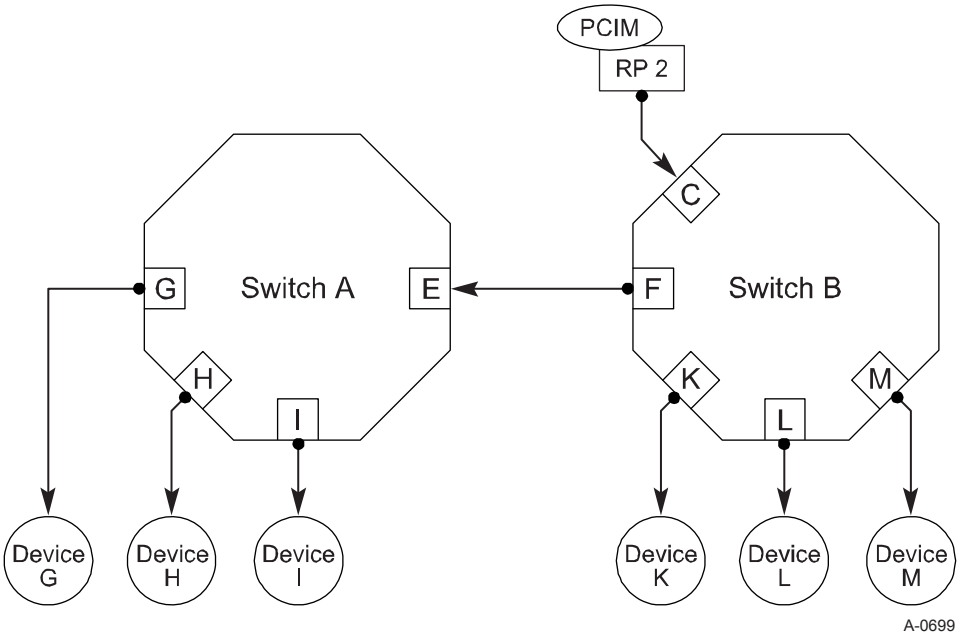

Figure 3-6: Example MR Topology: RP 2 View

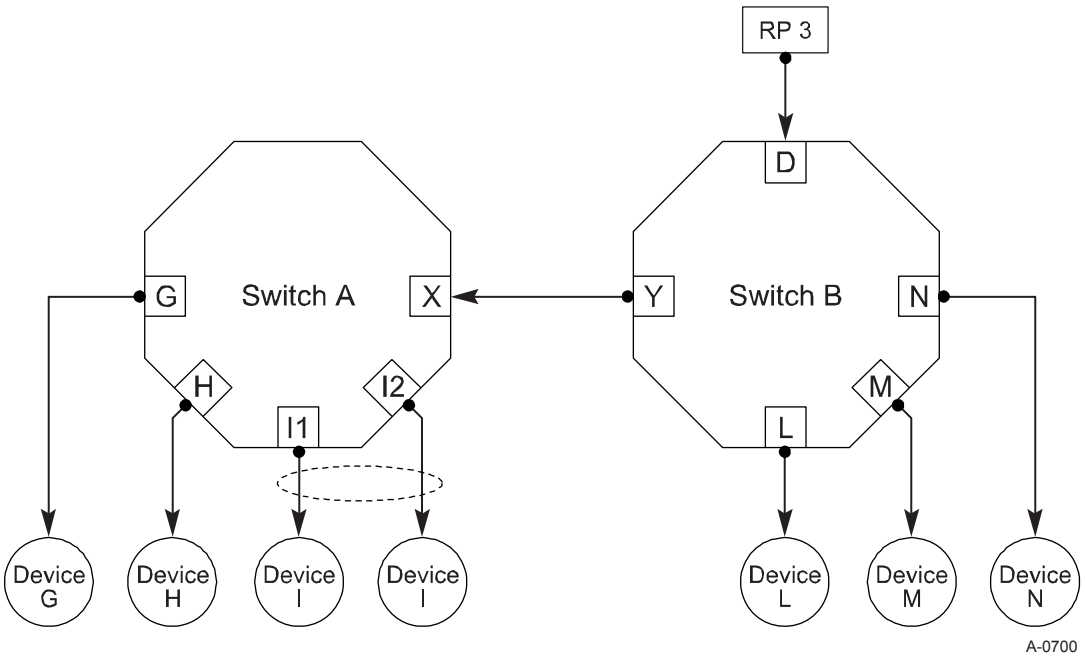

Figure 3-7: Example MR Topology: RP 3 View

Table 3-3: Example Topology: Switch A VS Bridge Table Contents

Slot	Root	VS	Bridge	Enabled	Mapped	Port	Port VHN
0	RP 0	VS 0	Upstream	Yes	Yes	A	VH 0 (PCIe)
1	RP 0	VS 0	Downstream 0	Yes	Yes	G	VH 0
2	RP 0	VS 0	Downstream 1	Yes	Yes	H	VH 0
3	RP 0	VS 0	Downstream 2	Yes	Yes	I	VH 0
4	RP 0	VS 0	Downstream 3	Yes	Yes	E	VH 0
5		VS 0	Downstream 4	No			
6		VS 0	Downstream 5	No			
7		VS 0	Downstream 6	No			
8		VS 0	Downstream 7	No			
9	RP 1	VS 1	Upstream	Yes	Yes	B	VH 0 (PCIe)
10	RP 1	VS 1	Downstream 0	Yes	Yes	G	VH 2
11	RP 1	VS 1	Downstream 1	Yes	No		
12	RP 1	VS 1	Downstream 2	Yes	Yes	H	VH 1
13	RP 1	VS 1	Downstream 3	Yes	Yes	J	VH 0 (PCIe)
14	RP 1	VS 1	Downstream 4	Yes	No		
15	RP 1	VS 1	Downstream 5	Yes	No		
16	RP 1	VS 1	Downstream 6	Yes	No		

Slot	Root	VS	Bridge	Enabled	Mapped	Port	Port VHN
17	RP 1	VS 1	Downstream 7	Yes	No		
18	RP 2	VS 2	Upstream	Yes	Yes	E	VH 1
19	RP 2	VS 2	Downstream 0	Yes	Yes	G	VH 1
20	RP 2	VS 2	Downstream 1	Yes	No		
21	RP 2	VS 2	Downstream 2	Yes	Yes	H	VH 2
22	RP 2	VS 2	Downstream 3	Yes	Yes	I	VH 1
23		VS 2	Downstream 4	No			
24		VS 2	Downstream 5	No			
25		VS 2	Downstream 6	No			
26		VS 2	Downstream 7	No			
27	RP 3	VS 3	Upstream	Yes	Yes	X	VH 0
28	RP 3	VS 3	Downstream 0	Yes	Yes	G	VH 3
29	RP 3	VS 3	Downstream 1	Yes	Yes	H	VH 3
30	RP 3	VS 3	Downstream 2	Yes	Yes	I	VH 3[11]
31	RP 3	VS 3	Downstream 3	Yes	Yes	I	VH 3[12]
32		VS 3	Downstream 4	No			
33		VS 3	Downstream 5	No			
34		VS 3	Downstream 6	No			
35		VS 3	Downstream 7	No			

Table 3-4: Example Topology: Switch B VS Bridge Table Contents

Slot	Root	VS	Bridge	Enabled	Mapped	Port	Port VHN
0	RP 2	VS 0	Upstream	Yes	Yes	C	VH 0 (PCIe)
1	RP 2	VS 0	Downstream 0	Yes	Yes	F	VH 1
2	RP 2	VS 0	Downstream 1	Yes	Yes	L	VH 0
3	RP 2	VS 0	Downstream 2	Yes	Yes	M	VH 0
4	RP 2	VS 0	Downstream 3	Yes	Yes	K	VH 0 (PCIe)
5		VS 0	Downstream 4	No			
6		VS 0	Downstream 5	No			

[11] Port I1 in Figure 3-7. Two PFs of Device I are assigned to RP 3. As far as RP 3 is concerned, these are independent Devices.

[12] Port I2 in Figure 3-7.

Slot	Root	VS	Bridge	Enabled	Mapped	Port	Port VHN
7		VS 0	Downstream 6	No			
8		VS 0	Downstream 7	No			
9	RP 0	VS 1	Upstream	Yes	Yes	F	VH 0
10	RP 0	VS 1	Downstream 0	Yes	Yes	L	VH 1
11	RP 0	VS 1	Downstream 1	Yes	Yes	M	VH 1
12	RP 0	VS 1	Downstream 2	Yes	No		
13	RP 0	VS 1	Downstream 3	Yes	No		
14	RP 0	VS 1	Downstream 4	Yes	No		
15	RP 0	VS 1	Downstream 5	Yes	No		
16	RP 0	VS 1	Downstream 6	Yes	No		
17	RP 0	VS 1	Downstream 7	Yes	No		
18	RP 3	VS 2	Upstream	Yes	Yes	D	VH 0 (PCIe)
19	RP 3	VS 2	Downstream 0	Yes	Yes	L	VH 2
20	RP 3	VS 2	Downstream 1	Yes	Yes	M	VH 2
21	RP 3	VS 2	Downstream 2	Yes	Yes	N	VH 0 (PCIe)
22	RP 3	VS 2	Downstream 3	Yes	Yes	Y	VH 0
23	RP 3	VS 2	Downstream 4	Yes	No		
24	RP 3	VS 2	Downstream 5	Yes	No		
25		VS 2	Downstream 6	No			
26		VS 2	Downstream 7	No			
27		VS 3	Upstream	No			
28		VS 3	Downstream 0	No			
29		VS 3	Downstream 1	No			
30		VS 3	Downstream 2	No			
31		VS 3	Downstream 3	No			
32		VS 3	Downstream 4	No			
33		VS 3	Downstream 5	No			
34		VS 3	Downstream 6	No			
35		VS 3	Downstream 7	No			

3.1.6.2. Example Topology: Device Implementation

In addition to Switch mapping, MR Devices need configuration.

Devices G and H are both assigned NumVHs of 4. No additional configuration is necessary. Switch configuration established the mapping between VH numbers and RPs.

Device I is assigned NumVHs of 6. No additional configuration is necessary. VH 4 and VH 5 of the Device could be dynamically assigned to unmapped Switch Ports associated with RP 0, RP 1, or RP 2 (RP 3 has no unmapped Switch Ports on Switch A).

Device L is assigned NumVHs of 3. VF Mapping is shown in Figure 3-8.

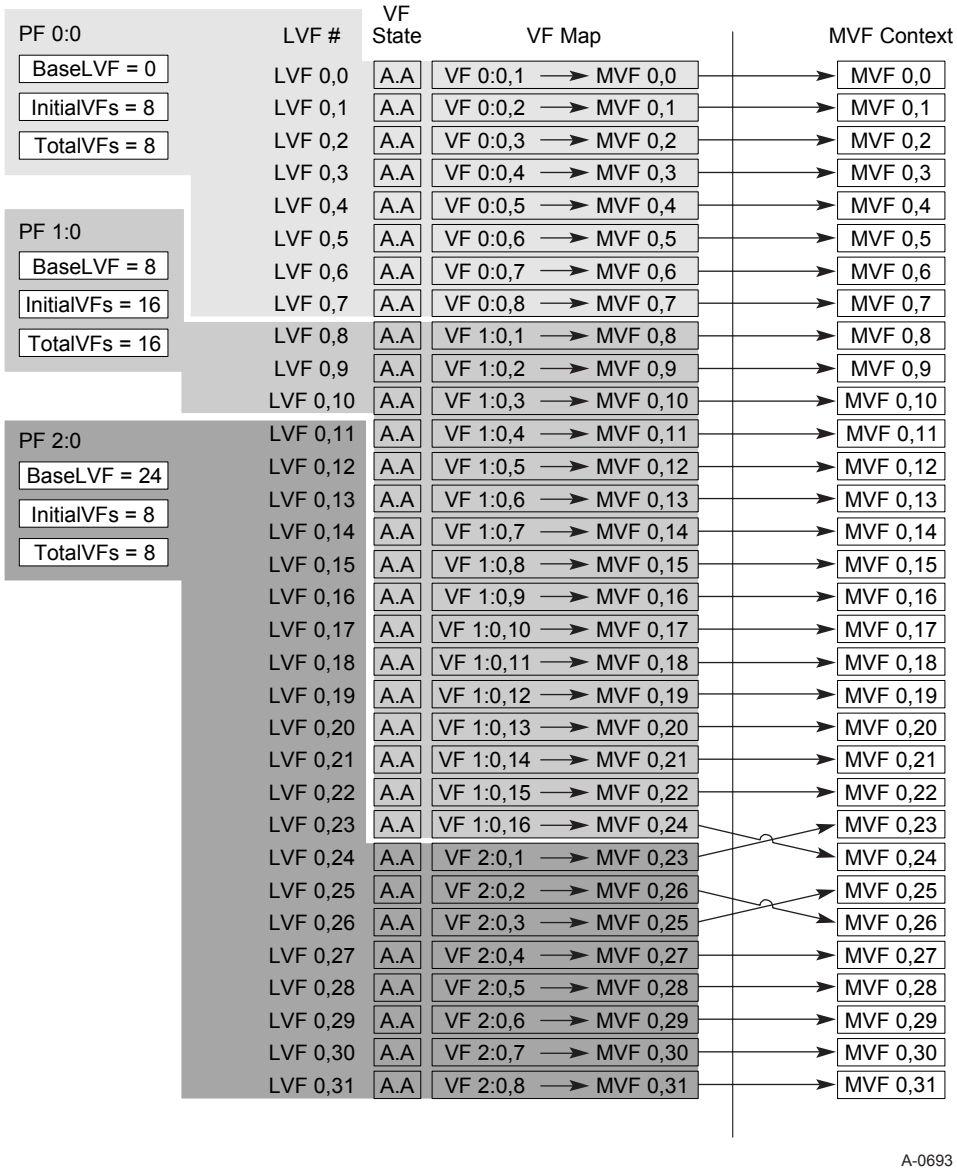

Figure 3-8: Example MR Topology: Device L PF/VF Mapping

Device M is assigned NumVHs of 3. VF Mapping is shown in Figure 3-9. VF Migration Capable is Set.

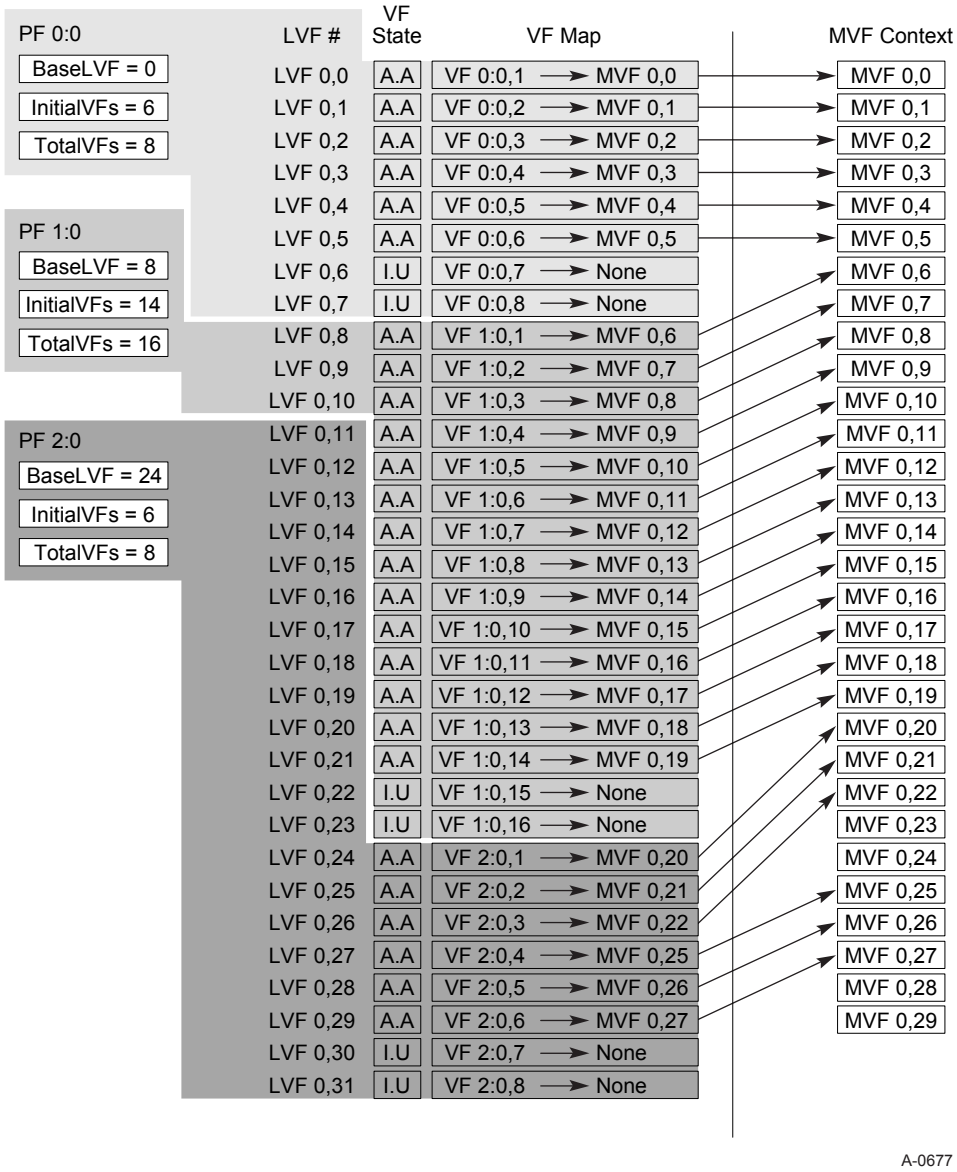

Figure 3-9: Example MR Topology: Device M VF Mapping

3.1.7. MR-PCIM Failover

There is a single active MR-PCIM in a topology. This MR-PCIM receives all "route to PCIM" errors and events. The VS associated with this MR-PCIM is programmed into each Switch's Management VS register.

There may be multiple Authorized VSs. Software running in these VSs is allowed to view and change the Switch MR-IOV configuration. To avoid confusion, coordination is needed between such software but is not specified by this specification.

A "backup" MR-PCIM operating in any Authorized VS may become the active MR-PCIM simply by changing the Management VS register in the affected Switch(s). This feature can support failover

from one MR-PCIM to a backup MR-PCIM. The Suppress Reset Propagation feature of a VS can be used to prevent a reset due to the failure of one MR-PCIM resetting state needed to continue operation using a backup MR-PCIM.

Mechanisms used to detect MR-PCIM failure, to select the new MR-PCIM location, to initiate the failover, etc., are undefined by this specification.

TLPs targeting MR-PCIM are regular PCIe TLPs within the appropriate VH. Such TLPs are forwarded based on the Switch configuration when they are received. Old TLPs are not re-routed due to Switch reconfiguration or change in VS Authorization.

3.2. MR Device Initialization

After Conventional Reset, MR Devices negotiate to use the MR Link protocol. If this negotiation is successful, the Device appears as VH 0 and contains one or more Base Functions (BFs) and zero or more PFs or Non-IOV Functions. Each BF contains an MR-IOV Capability in the Type 0 Configuration header.

Additional VHs beyond VH0 are enabled using the MR-IOV Capability.

Every BF is associated with a single PF or Non-IOV Function in each non-zero VH.

Every BF is optionally associated with a single PF or Non-IOV Function in VH 0.

Every PF (in any VH) and every Non-IOV Function (in any non-zero VH) is associated with a single BF.

Attached to each PF in each VH is an optional collection of Virtual Functions also in that VH. These are described in the *Single Root I/O Virtualization and Sharing Specification*.

PFs, BFs, and VFs are designated as follows:

BF *f* indicates the Base Function at Function number ***f*** (***f*** must be between 0 and 255).

PF ***f*** indicates a PF at Function number ***f*** (***f*** must be between 0 and 255). This nomenclature is used for SR systems or for the SR view of a single MR VH.

PF ***h:f*** indicates a PF within an MR system at Function number ***f*** in **VH** ***h*** (***h*** must be between 0 and the maximum VH number in use on the Link).

VF ***f,s*** indicates VF number *s* attached to PF number ***f*** (***s*** must be between 1 and the number of VFs in use for PF ***f***). This nomenclature is used for SR systems or for the SR view inside a single VH.

VF ***h:f,s*** indicates VF number ***s*** attached to PF number ***f*** in **VH** ***h***.

In addition, the optional VF Mapping and VF Migration features use the terms Logical Virtual Function (LVF) and Mission Virtual Function (MVF). These are similarly designated as follows:

LVF ***f,s*** indicates LVF table slot number ***s*** attached to PF number ***f***.

MVF ***f,s*** indicates MVF number ***s*** attached to PF number ***f***. MVFs do not have a VH (a VH is associated with the LVF that an MVF is mapped to as described below).

During topology enumeration, MR-PCIM detects MR Devices by noticing the presence of an MR-IOV Capability in PF 0's Configuration header.

Initializing and managing a Device in MR mode involves managing four aspects of the Device.

- Configuring and enabling the VHs.
- Enabling and managing the optional MR flow control features: This involves configuring the number of Virtual Links used, configuring the number of VCs offered in each VH, configuring the (VH, VC) to VL mapping hardware, configuring any VH to VL arbitration hardware, and configuring any VL to Link arbitration hardware.
- Enabling and managing the optional VF Mapping features: This involves configuring the number of LVFs offered by each PF in each VH and configuring each PF's LVF to MVF map.
- Enabling and managing the optional VF Migration features: This involves leaving "holes" in the LVF to MVH map to support migration, enabling VF Migration, responding to requests for initiate VF migration and interacting with SR-PCIM software to accomplish VF migration.

These aspects will be described separately in the following sections.

3.2.1. Enabling MR Operation

Initially MR Devices always negotiate to use the MR Link Protocol. If this negotiation fails, they operate in PCI Express mode. Initially, MR Devices operating in MR mode use only VH 0. VH 0 contains one or more Functions (PFs, BFs, or non-IOV Functions) operating as either a PCI Express single Function or Multi-Function Device.

In MR Devices, each BF contains an MR-IOV Capability block. There are a few key registers in this Capability used in enabling MR Operation.

- The MR-IOV Capabilities register indicates which optional features are implemented by the Device.
- One of the BFs is designated the Main BF. This BF contains certain fields that apply to the entire Device. These fields are reserved in other BFs (if any). The Main BF is identified by the "Is Main BF" bit in the MR-IOV Capability. The function number associated with the Main BF is Device Specific.
- The MaxVH register in the Main BF indicates the number of VHs supported by Device hardware. The value is Vendor Specific and must not change except after Fundamental Reset of the Device.
- The NumVH register in the Main BF indicates to the Device how many VHs are going to be used by MR-PCIM. Software should set this value based on the value of MaxVH, on the number of VHs implemented at the upstream end of the Link and on the number of VHs needed by the system. Once software has Set MR Enable, the NumVH value may not be changed.
- The MR-IOV Capability of every BF contains a pointer to the Function Table. This table contains one entry for every Function associated with the BF. This table is indexed by VH number since every BF contains a single function in each VH (Exception: VH 0 need not have a

function so the first Function Table entry might not be used (See the Function Present field in Section 4.2.4.1).

- The Function Table entry for the Main BF contains a VC ID to VL Map. This map includes a Map Enable bit. A VH is considered Enabled when, for some VC, the VC ID to VL Map entry is Enabled, points to a VL that is Enabled, and software operating in the VH enables that VC. See Section 4.2.4.4 for additional details.

3.2.2. Managing Flow Control

There are a number of fields used to manage flow control and VC to VL mapping.

MR Flow Control uses the same concepts as PCIe flow control. TLPs can only be sent if the transmitter has sufficient available flow control credits of the appropriate flavor. The differences from PCIe include:

In MR, VCs are replaced by Virtual Links (VLs). All Flow Control information is related to VLs not VCs. VCs continue to have meaning within a VH and serve as the mechanism used to present software operating in each VH with the collection of flow control channels and the mechanism that allows designating which TLPs should get to use each flow control channel. Software operating in a VH controls the TC to VC map. MR-PCIM controls the subsequent map from (VH, VC) to VL. Performance characteristics of a VL are assigned by MR-PCIM.

If **PF *h:f*** supports more than one VC, the either optional VC Capability or optional MFVC Capability exists in the PF ***h:f*** Type 0 Configuration header. The values presented are managed by MR-PCIM as follows:

- The **PF *h:f*** Extended VC Count value is configured by MR-PCIM. This value indicates to software operating in the VH the number of VC Resources that it can use.
- The **PF *h:f*** Low Priority Extended VC Count value is configured by MR-PCIM. This value indicates to software operating in the VH which VC Resources should be viewed as higher priority.
- The **PF *h:f*** VC to VL map is configured by MR-PCIM. This map converts VC IDs chosen by software running in the VH into the corresponding VL numbers. If more than one VC is supported, this is a full eight entry map since all VC ID values are legal. This map also includes an Enable bit.
- The **PF *h:f*** VC Enabled fields allow MR-PCIM to determine which VC Resources are enabled by software operating in the VH. When VC Enabled changes value, the VC Config Changed interrupt is signaled to MR-PCIM.
- The **PF *h:f*** VC ID fields allow MR-PCIM to determine the VC ID assignments made by software operating in the VH.
- The **PF *h:f*** TC to VC map field allows MR-PCIM to determine the mapping between TCs and VCs made by software operating in the VH.

3.2.3. Managing VF Mapping

Software operating in the VH (e.g., SR-PCIM) can optionally see a collection of Virtual Functions attached to each PF. In MR, these VFs are known as Logical Virtual Functions (LVFs). MR Devices implement some number of underlying Mission Virtual Functions (MVFs). These MVFs are mapped into LVFs. This mapping can be optionally controlled by MR-PCIM using VF Mapping hardware. This VF Mapping support is optional. If VF Mapping is not supported for a BF, LVFs are still mapped to MVFs but the mapping is Device Specific and is not visible or controllable by MR-PCIM.

VF Mapping uses a set of mapping tables controlled by management software. These tables allow software to (1) control the number of LVFs assigned to each PF, (2) specify which MVFs (if any) are assigned to each LVF, and (3) detect how many VFs software operating in the VH has indicated it will use.

Figure 3-10 shows an example setup. There are 5 VHs, each with a single PF at Function 0. VH 1 through VH 4 have been assigned some VFs.

- VFs in VH 1 have been assigned 8 LVFs and 4 MVFs. LVF 0-0 through LVF 0-7 are associated with the 8 VFs. There are 4 VF holes (VF 1:0.5 through VF 1:0.8 a.k.a. LVF 0-4 through LVF 0-7) meaning that, if SR-PCIM enables VF Migration, up to 4 MVFs can be migrated in to VH 0.
- VFs in VH 2 have been assigned 4 LVFs and 4 MVFs. All LVFs are populated meaning that no migration in is possible unless an MVF first migrates out.
- VFs in VH 3 have been assigned 7 LVFs and 5 MVFs. Like VH 0, holes were left for possible VF Migration in to the VH.
- VFs in VH 4 have been assigned 3 LVFs and 3 MVFs. Like VH 1, no migration in is possible unless an MVF first migrates out.
- No VFs have been assigned to VH 5.

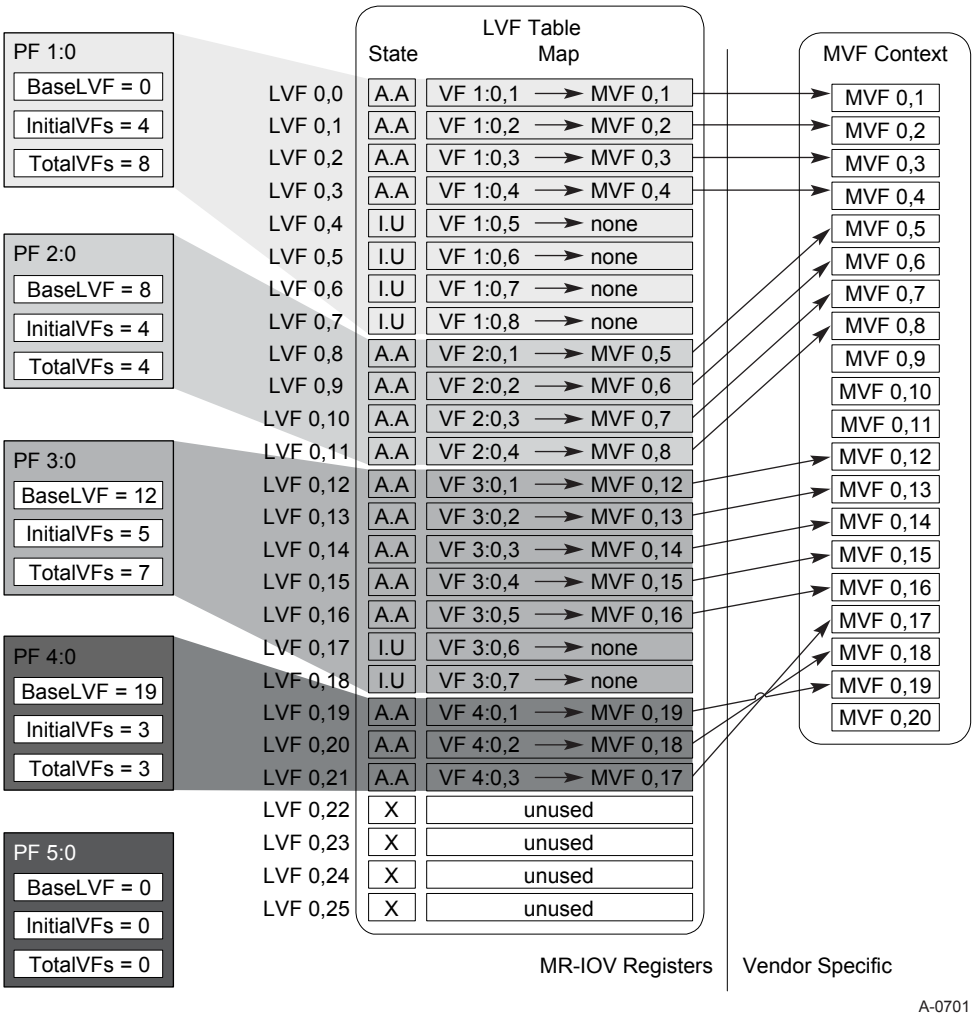

Figure 3-10: Example Mapping of VFs

The VF State table is used to coordinate migration of MVFs in/out of LVFs. If hardware supports VF Mapping but not VF Migration, the VF State table is Read Only Zero. If hardware supports VF Migration, management software must configure the VF State in the LVF Table even if the VF Migration Capable bit is Clear. The VF State values shown in Figure 3-10 are the required initial values. See Section 3.2.4 for details.

MR-PCIM manages the LVF slot allocation to PFs using the Base LVF and Total VFs registers. Base LVF indicates the first LVF slot associated with the PF. Base LVF + Total LVFs indicates the last LVF slot associated with the PF. The LVF slot designated by Base LVF contains the MVF associated with the PF. The LVF slot designated by Base LVF+1 contains the MVF associated with VF *x*.1 of PF *x*, etc. Every PF has at least one LVF slot.

The number of populated LVFs offered to SR-PCIM is contained in InitialVFs. If SR-PCIM does not enable VF Migration, only these slots are used and any additional unpopulated slots remain unused.

Initially, MR-PCIM must populate LVFs for a given PF using the lower numbered VFs first. Holes left for migration (if any) follow the last populated LVF.

Before it enables SR-IOV operation, SR-PCIM in each VH must set the NumVFs value to indicate the number of VFs it wishes to use. The value set must be less than or equal to TotalVFs. If VF Migration not enabled by SR-PCIM, the value set must also be less than or equal to InitialVFs.

In some Devices, the Device Programming Model assumes that operations on one Function can affect another Function of the Device. In MR, this dependency manifests itself as a relationship between MVFs of some of the Device's BFs. The Function Dependency Link field in a BF indicates the presence a dependency. See Section 4.2.1.2 for details. A similar dependency exists in the *Single Root I/O Virtualization and Sharing Specification*, but there it deals with dependencies associated with VF assignment to SIs.

3.2.4. Managing VF Migration

In Multi-Root systems, VFs can be migrated between VHs. VF Migration does not occur in Single-Root only systems.

SR-IOV and MR-IOV support for VF Migration is optional. VF Migration for VFs associated with a BF is only possible if all three of the following are true:

- The VF Migration Supported bit in the BF MR-IOV Capability is Set.
- In at least one VH (***h***), MR-PCIM software has set the VF Migration Capable bit in the Function Table entry controlling **PF *h:f***.
- In that VH (***h***), SR-PCIM software controlling **PF *h:f*** has also Set the VF Migration Enabled bit.

VF Migration centers around the LVF Table:

- The VF State field manages the SR-IOV and MR-IOV combined view of the migration state of a VF. This table is used to gracefully add and remove VFs to or from VHs.
- The VF Map field maps LVFs onto MVFs. When the VF State is Inactive.Unavailable, software can write this field to implement a change.

VF migration follows the state diagram shown in Figure 3-11. The state values shown are contained in the VF State field associated with the VF. State transitions indicated by solid lines are initiated by MR software by writing the new state value to the VF State field. State transitions indicated by dashed lines are typically initiated by SR-PCIM and are visible to MR-PCIM via the VF State field. The mechanisms used for this are described in the *Single Root I/O Virtualization and Sharing Specification*.

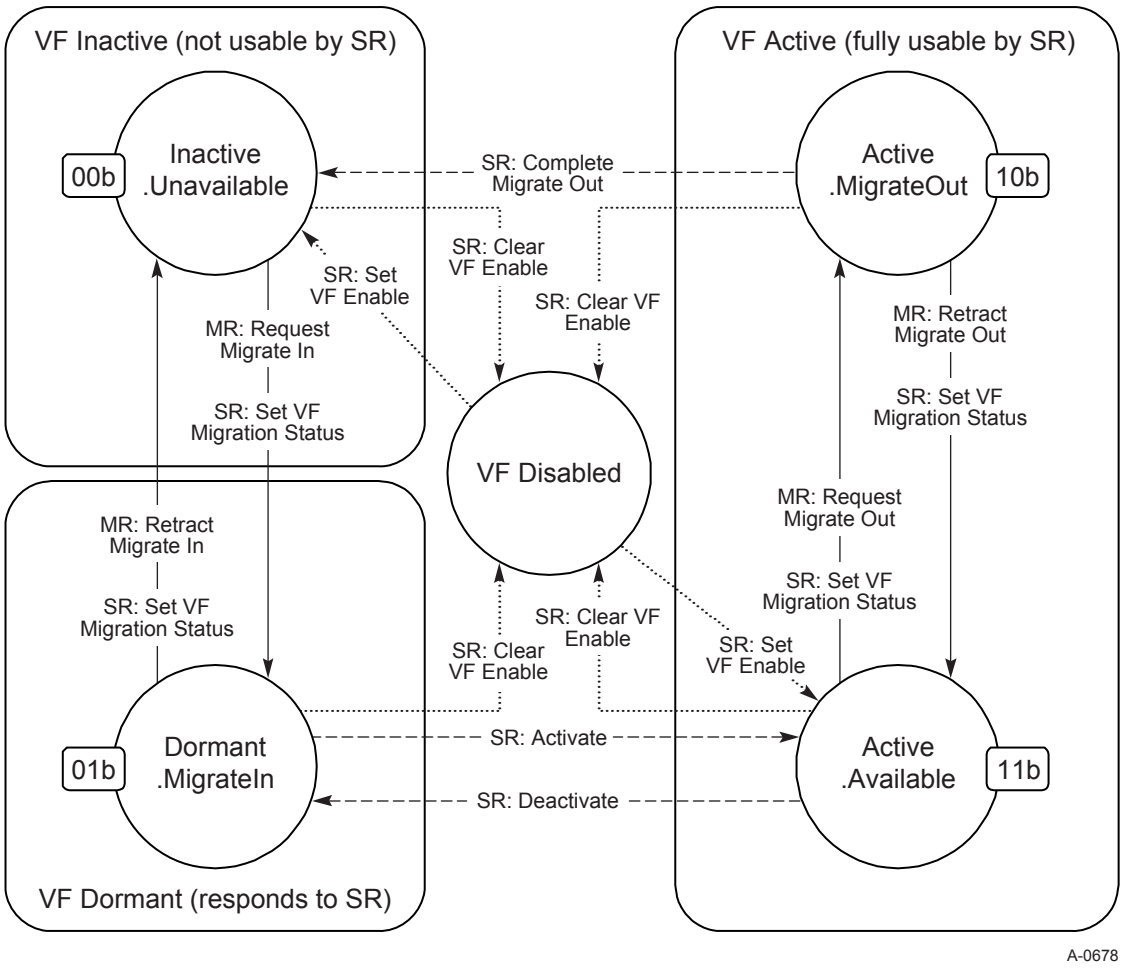

Figure 3-11: VF Migration State Diagram

The following state transitions may be initiated by MR software by writing the VF State field. Any other writes are ignored and no state transition occurs.

Table 3-5: Valid MR State Transitions for VF Migration

Current State		Written State		Meaning
00b	Inactive.Unavailable	01b	Inactive.MigrateIn	Request Migrate In
01b	Inactive.MigrateIn	00b	Inactive.Unavailable	Retract Migrate In
11b	Active.Available	10b	Active.MigrateOut	Request Migrate Out
10b	Active.MigrateOut	11b	Active.Available	Retract Migrate Out

VFs that are in the Inactive.Unavailable state are not usable by software in the VH. Configuration, I/O, and Memory Requests within the VH targeting the associated VF return UR. Within 100 ms of transitioning to this state, a VF must stop issuing Requests.

VFs that are in the Inactive.MigrateIn state (1) will respond to Configuration Requests issued by software running in the VH, (2) if MSE is Set, will respond to Memory requests issued by software running in the VH, and (3) will not issue Requests.

The state transition from Active.MigrateOut to Inactive.Unavailable Sets the MR VF Migration Status bit. If the MR VF Migration Interrupt Enable is Set, this in turn causes an MSI to be queued to MR-PCIM. MR-PCIM software can then scan the LVF Table to determine the cause of the interrupt. Specifically, MR-PCIM is looking for the VFs that it previously placed in the Active.MigrateOut state and are now in the Inactive.Unavailable state.

State transitions with the notation "SR: Set VF Migration Status" cause similar behavior within the VH. See the *Single Root I/O Virtualization and Sharing Specification* for details.

The following steps are used by MR-PCIM to migrate a VF from one VH to another.

1. Request arrives from higher level software requesting that **VF *h:f,s*** be migrated to **VF *a:f,c***. For the request to be valid, the VF Table Entry associated **VF *h:f,s*** must be in the Active.Available state and the VF Table Entry associated with **VF *a:f,c*** must be in the Inactive.Unavailable state.
2. Initiate a Migrate Out operation in **VH h** by writing the VF State entry associated with **VF h:f,s** to the Active.MigrateOut.
3. Wait for SR-PCIM to stop using the VF and to indicate so by transitioning the VF State to Inactive.Unavailable. This transition sets the MR VF Migration Status bit and can raise an interrupt to MR-PCIM.
4. Save the value of the VF Map entry associated with **VF *h:f,s***.
5. Set the VF Map entry associated with **VF *h:f,s*** to zero to indicate an empty slot.
6. Set the VF Map entry associated with **VF *a:f,c*** the value saved in step 4.
7. Initiate a Migrate In operation in **VH a** by writing the VF Map entry associated with **VF a:f,c** to the Inactive.MigrateIn state.
8. At some point, SR-PCIM will transition the VF the Active.Available state and start using it.

In addition to the graceful migration described above, MR-PCIM can retract a Migrate In or Migrate Out request that it previously requested.

3.2.4.1. VF Migration Initial State

Software in a VH is expecting to see the initial VF configuration shown in Figure 3-12. MR-PCIM must ensure this condition by appropriate programming of the VF Migration tables.

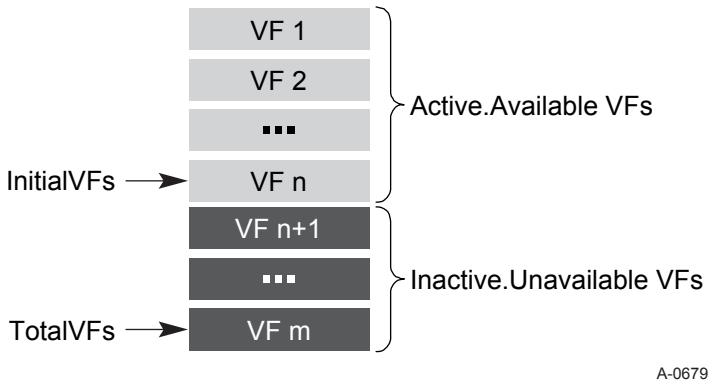

Figure 3-12: Initial VF State

Specifically, for each PF, MR-PCIM must configure the associated Function Table entry such that:

- InitialVFs ≥ 0
- TotalVFs ≥ InitialVFs
- 0 ≤ BaseLVF + TotalVFs < MaxLVF
- The LVF table region assigned to the PF, [BaseLVF **..** BaseLVF + TotalVFs – 1], does not overlap with the region assigned to any other PF of the BF.
- If InitialVFs > 0, all LVF entries in the range [BaseLVF **..** BaseLVF + InitialVFs – 1] are in state Active.Available and are mapped to valid MVFs.
- If InitialVFs != TotalVFs, all LVF entries in the range [BaseLVF + InitialVFs **..** BaseLVF + TotalVFs – 1] are in state Inactive.Unavailable.

3.2.4.2. VF Migration Reinitialization

After a PF is Reset or when VF Enable is Cleared and then Set, a valid initial VF configuration must be re-established. The InitialVFs value may be different from an earlier initial configuration so long as the configuration meets the rules described in Section 3.2.4.1.

This process can be accomplished by hardware or software and must be completed within 1 s (to avoid an SR software timeout resulting in the hardware being declared broken).

This process starts by adjusting InitialVFs to reflect the number of active VFs associated with the PF and then rearranging those active VFs into lower numbered VFs, keeping the same relative VF ordering.

IMPLEMENTATION NOTE

SR-IOV Interaction

As described in the *Single Root I/O Virtualization and Sharing Specification*, if VF Migration Enable was Set, SR software must wait 1 second after clearing VF Enable for fields in the SR-IOV Capability to become valid (including InitialVFs and TotalVFs).

3.3. MR Root Port Initialization

MR Root Complexes consist of one or more traditional Root Complex each with one or more Root Ports and a Vendor Specific mapping of those Root Ports into one or more MR Links.

For each MR Link, one Root Port is mandatory and is associated with VH 0 on that Link. This Root Port controls the physical Link. When the Link operates in MR Mode, additional Root Ports are optional and are associated with additional VHs of the MR Link.

Note that on MR Links, only VH 0 is initially available and non-zero VHs become available when enabled by software. As such, Root Ports associated with non-zero VHs are not initially usable.

For Links that are MR Enabled, the RP sends MRInit DLLPs as shown in Table 2-1 with the Device/Port Type indicating Root Port (0100b).

The mechanisms used to manage MRA Root Ports are vendor specific and outside the scope of this specification.

4

4. Configuration

The following sections list the configuration requirements for Base Function (BF), Physical Functions, MRA Switches, and MR-PCIMs.

Memory Space tables defined in this chapter must support naturally aligned DWORD and QWORD access. Access of other sizes or alignments is undefined.

If a BAR that maps address space for structures defined in this specification also maps other usable address space not defined in this specification, locations used in the other address space must not share any naturally aligned 8-KB address range with one where structures defined in this specification reside.

4.1. Configuration Field Summary

The following fields are used to manage MR IOV features of a Device or Switch.

Table 4-1: MR-IOV Fields

Field Name	Width	Type	Opt	Device Usage	Switch Usage
MR Enable	1	RW	Req	Main BF/Control	n/a
MSI Vector #	11	RO	Req	BF/Capability	MR-IOV Cap/Capability
MR Switch Number	16	RW	Req	n/a	MR-IOV Cap/Control
Function Dependency Link	8	RO	Req	BF/Capability	n/a
VS Interrupt Enable	1	RW	Req	n/a	MR-IOV Cap/Control
Port Interrupt Enable	1	RW	Req	n/a	MR-IOV Cap/Control
VS Interrupt Status	1	RO	Req	n/a	MR-IOV Cap/Status
Port Interrupt Status	1	RO	Req	n/a	MR-IOV Cap/Status
VS #	8	RO	Req	n/a	MR-IOV Cap/Status
VS Bridge #	8	RO	Req	n/a	MR-IOV Cap/Status
VS is Authorized	1	RO	Req	n/a	MR-IOV Cap/Status
# VS	8	RO	Req	n/a	MR-IOV Cap/Capability
# VS Bridge	8	RO	Req	n/a	MR-IOV Cap/Capability
# Port	8	RO	Req	n/a	MR-IOV Cap/Capability
VS Table Entry Size	8	RO	Req	n/a	MR-IOV Cap/Capability
VS Table Offset/BIR	32	RO	Req	n/a	MR-IOV Cap
VS Bridge Table Entry Size	8	RO	Req	n/a	MR-IOV Cap/Capability
VS Bridge Table Offset/BIR	32	RO	Req	n/a	MR-IOV Cap
Port Table Entry Size	8	RO	Req	n/a	MR-IOV Cap/Capability
Port Table Offset/BIR	32	RO	Req	n/a	MR-IOV Cap
Function Present	1	RO	Req	BF/Capability	n/a
Function Offset	8	RO	Req	BF/Capability	n/a
Watchdog Timer Support					
Watchdog Timer Interrupt Enable	1	RW	Req	n/a	MR-IOV Cap/Control

Field Name	Width	Type	Opt	Device Usage	Switch Usage
Watchdog Timer Interrupt Status	1	RW1C	Req	n/a	MR-IOV Cap/Status
Timer Interval 1	8	RW	Req	n/a	MR-IOV Cap/Watchdog
Timer Interval 2	8	RW	Req	n/a	MR-IOV Cap/Watchdog
Watchdog 1 Expired	1	RW1C	Req	n/a	MR-IOV Cap/Watchdog
Rearm Watchdog 1 & 2	1	RW1C	Req	n/a	MR-IOV Cap/Watchdog
Performance Monitoring					
Statistics Interrupt Enable	1	RW	Opt	BF/Control	MR-IOV Cap/Control
Statistics Interrupt Status	1	RW1C	Opt	BF/Status	MR-IOV Cap/Status
# Statistics Blocks	8	RO	Req	BF/Statistics	MR-IOV Cap/Statistics
# Statistics Descriptors	8	RO	Req	BF/Statistics	MR-IOV Cap/Statistics
Statistics Block Start/Busy	up to 32	RW	Opt	BF/Statistics	MR-IOV Cap/Statistics
Statistics Block Offset/BIR	32	RO	Opt	BF/Statistics	MR-IOV Cap/Statistics
Statistics Descriptor Offset/BIR	32	RO	Opt	BF/Statistics	MR-IOV Cap/Statistics
Link Control					
MaxVH	8	RO	Req	Main BF/VH Counts	Port Table/Capability
NumVH	8	RW	Req	Main BF/VH Counts	Port Table/Control
Port Present	1	RO	Req	n/a	Port Table/Capability
Port Enable	1	RW	Req	n/a	Port Table/Control
Port Interrupt Enable	8	RW	Req	n/a	Port Table/Control
Port Interrupt Pending	8	RW1C	Req	n/a	Port Table/Status
MR-IOV Link Enable	1	RW	Req	n/a	Port Table/Control
Force Reset	1	RW	Req	n/a	VS Bridge Table/Hot Plug Signals
VS Interrupt Enable	1	RW	Req	n/a	VS Table/Control
VS Suppress Reset Propagation	1	RW	Req	n/a	VS Table/Control
PM_PME Triggers Beacon/WAKE#	1	RW	Opt	n/a	Port Table/Control

Field Name	Width	Type	Opt	Device Usage	Switch Usage
Send PME_Enter_L23 DLLP	1	RW	Opt	n/a	Port Table/Control
Non-PCIe Management Port	1	RO	Req	n/a	Port Table/Capability
MR Error Status	32	RW1C	Opt	Main BF/MR Log	Port Table/MR Log
MR Header Log	160 (5*32)	RO	Opt	Main BF/MR Log	Port Table/MR Log
Max Payload Size Supported	3	RO	Req	n/a	VS Bridge Table/Capability
Max Payload Size Offered	3	RW	Opt	n/a	VS Bridge Table/Control
Link Direction Status	2	RO	Req	n/a	Port Table/Status
Link Partner Detected	1	RO	Req	n/a	Port Table/Status
Link Direction Control + Backup Link Direction Control	2+2	RW	Req	n/a	Port Table/Control
Link Partner MaxVH	8	RO	Req	n/a	Port Table/Link Partner
Link Partner MaxVL	3	RO	Req	n/a	Port Table/Link Partner
Link Partner Trained in MR Mode	1	RO	Req	n/a	Port Table/Link Partner
Link Partner Protocol Version	3	RO	Req	n/a	Port Table/Link Partner
Link Partner was Authorized	1	RO	Req	n/a	Port Table/Link Partner
Link Partner Device/Port Type	4	RO	Req	n/a	Port Table/Link Partner
Link Partner Mixed Device/Port Type	1	RO	Req	n/a	Port Table/Link Partner
Link Partner VH FC	1	RO	Req	n/a	Port Table/Link Partner
VF Migration/VF Mapping					
VF Migration Supported	1	RO	Req	BF/Capability	n/a
VF Mapping Supported	1	RO	Req	BF/Capability	n/a
VF Enable	1	RO	Req	Function Table/Status	n/a
VF Enable Changed	1	RW1C	Req	Function Table/Control	n/a
VF Enable Enabled	1	RW	Req	Function table/Control	n/a
VF Migration Capable	1	RW	Opt	Function Table/Control	n/a
VF Migration Enabled	1	RO	Opt	Function Table/Status	n/a
Max LVF #	16	RO	Opt	BF/VF Migration	n/a

Field Name	Width	Type	Opt	Device Usage	Switch Usage
Max MVF #	16	RO	Opt	BF/VF Migration	n/a
LVF Table Offset/BIR	32	RO	Opt	BF/VF Migration	n/a
Base LVF	16	RW	Opt	Function Table	n/a
InitialVFs	16	RW	Opt	Function Table/Control	n/a
TotalVFs	16	RW	Opt	Function Table/Control	n/a
NumVFs	16	RO	Opt	Function Table/Status	n/a
First VF Offset	16	RO	Opt	Function Table	n/a
VF Stride	16	RO	Opt	Function Table	n/a
VF Migration Status Interrupt Enable	1	RW	Opt	Function Table	n/a
VF Migration Status	1	RW1C	Opt	Function Table	n/a
PF Reset Indicated Interrupt Enable	1	RW1C	Opt	Function Table	n/a
PF Reset Indicated	1	RW1C	Opt	Function Table	n/a
Congestion Management					
VL Enable	8	RW	Req	Main BF/Control	Port Table/Control
VL Negotiation Pending	8	RO	Req	Main BF/Status	Port Table/Status
MaxVL	3	RW	Opt	BF/VL Arb	Port Table/VL Arb
BF VL	3	RW	Req	Main BF/Control	n/a
Default VL	3	RW	Opt	Main BF/Control	n/a
VL Arbitration Table Offset	30	RO	Opt	BF/VL Arb	Port Table/VL Arb
VL Arbitration Reference Clock	2	RO	Opt	BF/VL Arb	Port Table/VL Arb
VL Arbitration Capability	8	RO	Opt	BF/VL Arb	Port Table/VL Arb
Load VL Arbitration Table	1	RW	Opt	BF/VL Arb	Port Table/VL Arb
VL Arbitration Select	4	RW	Opt	BF/VL Arb	Port Table/VL Arb
VL Arbitration Status	1	RO	Opt	BF/VL Arb	Port Table/VL Arb
Max Time Slots	8	RO	Opt	BF/VL Arb	Port Table/VL Arb
VL Strict Priority Arbitration	8	RW	Opt	BF/VL Arb	Port Table/VL Arb
VC Capability Supported	1	RO	Req	Function Table/Capability	n/a
Num VC Resources Hardware Present	3	RO	Req	Function Table/Capability	n/a

Field Name	Width	Type	Opt	Device Usage	Switch Usage
Num MFVC Resources Hardware Present	3	RO	Req	Function Table/Capability	n/a
MFVC Capability Supported	1	RO	Req	Function Table/Capability	n/a
Extended VC Count	3	RW	Opt	Function Table/Control	VS Bridge Table/Control
Low Priority Extended VC Count	3	RW	Opt	Function Table/Control	VS Bridge Table/Control
VC Resource Enabled	8 (1*8)	RO	Opt	Main BF Function Table/VC State	VS Bridge Table/VC State
VC Resource VC Negation Pending	8 (1*8)	RO	Opt	Main BF Function Table/VS State	VS Bridge Table/VC State
VC ID	24 (3*8)	RO	Opt	Function Table/VC State	VS Bridge Table/VC State
TC to VC Map	64 (8*8)	RO	Opt	Function Table/VC State	VS Bridge Table/VC State
VC to VL Map	24 (3*8)	RW	Opt	Main BF Function Table/VC ID to VL Map	VS Bridge Table/VC ID to VL Map
VC Mapped	8 (1*8)	RW	Opt	Main BF Function Table/VC ID to VL Map	VS Bridge Table/VC ID to VL Map
VC Config Changed	1	RW1C	Opt	Main BF Function Table/Status	VS Bridge Table/Status
VC Config Interrupt Enable	1	RW	Opt	Main BF Function Table/Control	VS Bridge Table/Control
MFVC Resource Enabled	8 (1*8)	RO	Opt	Main BF Function Table/MFVC State	n/a
MFVC Resource VC Negation Pending	8 (1*8)	RO	Opt	Main BF Function Table/MFVC State	n/a
MFVC: VC ID	24 (3*8)	RO	Opt	Function Table/MFVC State	n/a
MFVC: TC to VC Map	64 (8*8)	RO	Opt	Function Table/MFCV State	n/a
MFVC Config Changed	1	RW1C	Opt	Main BF Function Table/Status	n/a
MFVC Config Interrupt Enable	1	RW	Opt	Main BF Function Table/Control	n/a

Field Name	Width	Type	Opt	Device Usage	Switch Usage
Global Key Management					
Global Key Value	12	RW	Req	Main BF/Function Table	VS Table/Control
Device Global Key Check Enable	1	RW	Req	Main BF/Control	n/a
VS Global Key Check Enable	3	RW	Req	n/a	VS Table/Control
Management Authorization					
Management VS	8	RW	Req	n/a	MR-IOV Cap/Control
Authorized VS Bitmap Offset	12	RO	Req	n/a	MR-IOV Cap/Control
Authorized VS Bitmap + Backup Authorized Bitmap	2 * NumVS	RW	Req	n/a	MR-IOV Cap/Control
Switch Mapping					
Bridge Hardware Present	1	RO	Req	n/a	VS Bridge Table/Capability
Bridge Enable	1	RW	Req	n/a	VS Bridge Table/Control
Bridge Port	8	RW	Req	n/a	VS Bridge Table/Control
Bridge Port VH	12	RW	Req	n/a	VS Bridge Table/Control
Port Mapped to Bridge	1	RW	Req	n/a	VS Bridge Table/Control
VS Present	1	RO	Req	n/a	VS Table/Capability
VS Enable	1	RW	Req	n/a	VS Table/Control
VS Interrupt Vector Num	11	RO	Req	n/a	VS Table/Capability
Hot Plug Signals Interface					
Hot Plug Hardware Present	1	RO	Req	n/a	VS Bridge Table/Capability
Bridge Controls Physical Link	1	RW	Req	n/a	VS Bridge Table/Hot Plug Signals
Slot Implemented	1	RW	Opt	n/a	VS Bridge Table/Hot Plug Signals
Virtual Hot Plug Interrupt Enable	1	RW	Opt	n/a	VS Bridge Table/Hot Plug Signals
PME Turn Off State	1	RO	Req	n/a	VS Bridge Table/Status

Field Name	Width	Type	Opt	Device Usage	Switch Usage
Physical Slot Number	13	RW	Opt	n/a	VS Bridge Table/Hot Plug Signals
Slot Power Limit Scale	2	RO	Opt	n/a	VS Bridge Table/Hot Plug Signals
Slot Power Limit Value	8	RO	Opt	n/a	VS Bridge Table/Hot Plug Signals
Hot Plug Capable	1	RW	Opt	n/a	VS Bridge Table/Hot Plug Signals
Hot Plug Surprise	1	RW	Opt	n/a	VS Bridge Table/hot Plug Signals
Attention Indicator State	2	RO	Opt	n/a	VS Bridge Table/Hot Plug Signals
Attention Indicator Changed	1	RW1C	Opt	n/a	VS Bridge Table/Status
Push Attention Button	1	RW	Opt	n/a	VS Bridge Table/Hot Plug Signals
Power Indicator State	2	RO	Opt	n/a	VS Bridge Table/Hot Plug Signals
Power Indicator Changed	1	RW1C	Opt	n/a	VS Bridge Table/Status
Power Controller State	1	RO	Opt	n/a	VS Bridge Table/Hot Plug Signals
Power Controller Changed	1	RW1C	Opt	n/a	VS Bridge Table/Status
Power Controller Present	1	RW	Opt	n/a	VS Bridge Table/Hot Plug Signals
Signal Power Fault	1	RW	Opt	n/a	VS Bridge Table/Hot Plug Signals
Presence Detect State	1	RW	Opt	n/a	VS Bridge Table/Hot Plug Signals

4.2. Device Configuration Space

For managing MR-IOV Devices, MR-PCIM uses MR-IOV Capabilities located in the Base Functions. Base Functions are Type 0 headers located in VH0 of the Device that contain the MR-IOV Capability.

Configuration controls are associated with:

- Entire component
- Each Base Function
- Each VH supported by the component

- Each Function or PF supported in some non-management VH by the component

There are up to six tables provided by each Base Function. These tables are located using the MR-IOV Capability block located in the Type 0 Configuration Space of the associated BF. An overview of the tables is show in Figure 4-1.

- The MR-IOV Capability contains information concerning the BF.
- The Function Table contains one entry for each Function or PF in each non-management VH. The Function Table associated with one designated "main" BF contains information concerning each VH supported by the Device.
- The optional VL Arbitration Table contains information describing how VL Arbitration is performed by the Device. If present, this table is associated with the BF that contains the VH Table.
- The optional LVF Table is used to control VF Mapping and VF Migration. This table is absent if neither VF Mapping nor VF Migration are supported by the BF.
- The Statistics Descriptor Table describes the Statistics gathering capabilities of the Device. This table is Read Only.
- The Statistics Block Table describes a collection of Statistics registers.
- The Statistics Table contains the various Statistics registers.

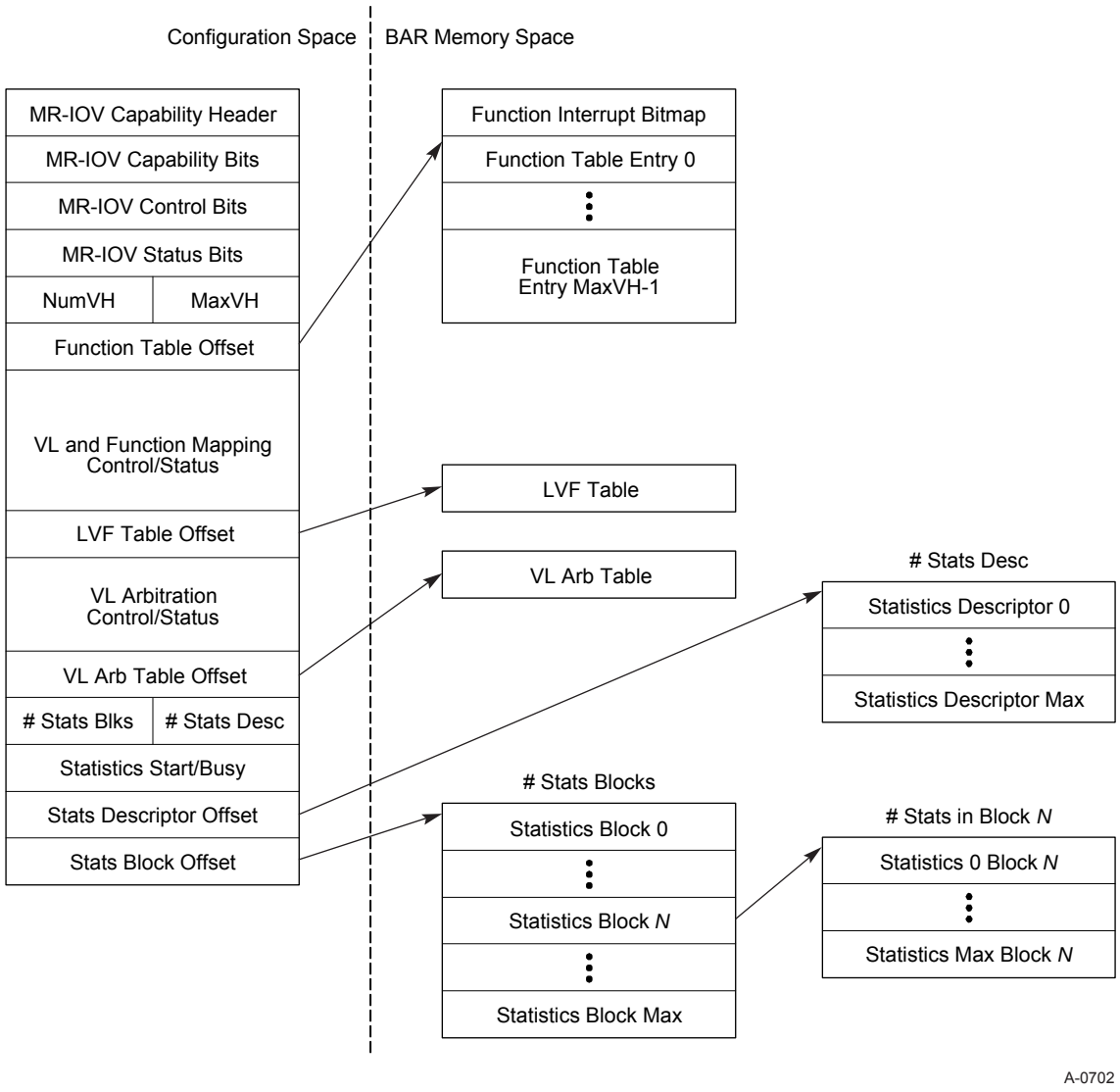

Figure 4-1: MR Device Configuration Space

4.2.1. Device MR-IOV Extended Capability

MR-IOV Base Functions contain a new PCIe Extended Capability. This capability is used by MR-PCIM to determine if a Device is MR-IOV capable and to manage the MR-IOV features of the Function.

The MR-IOV capability is located in the PCIe Extended Configuration Space. Figure 4-2 shows the Device MR-IOV Extended Capability structure:

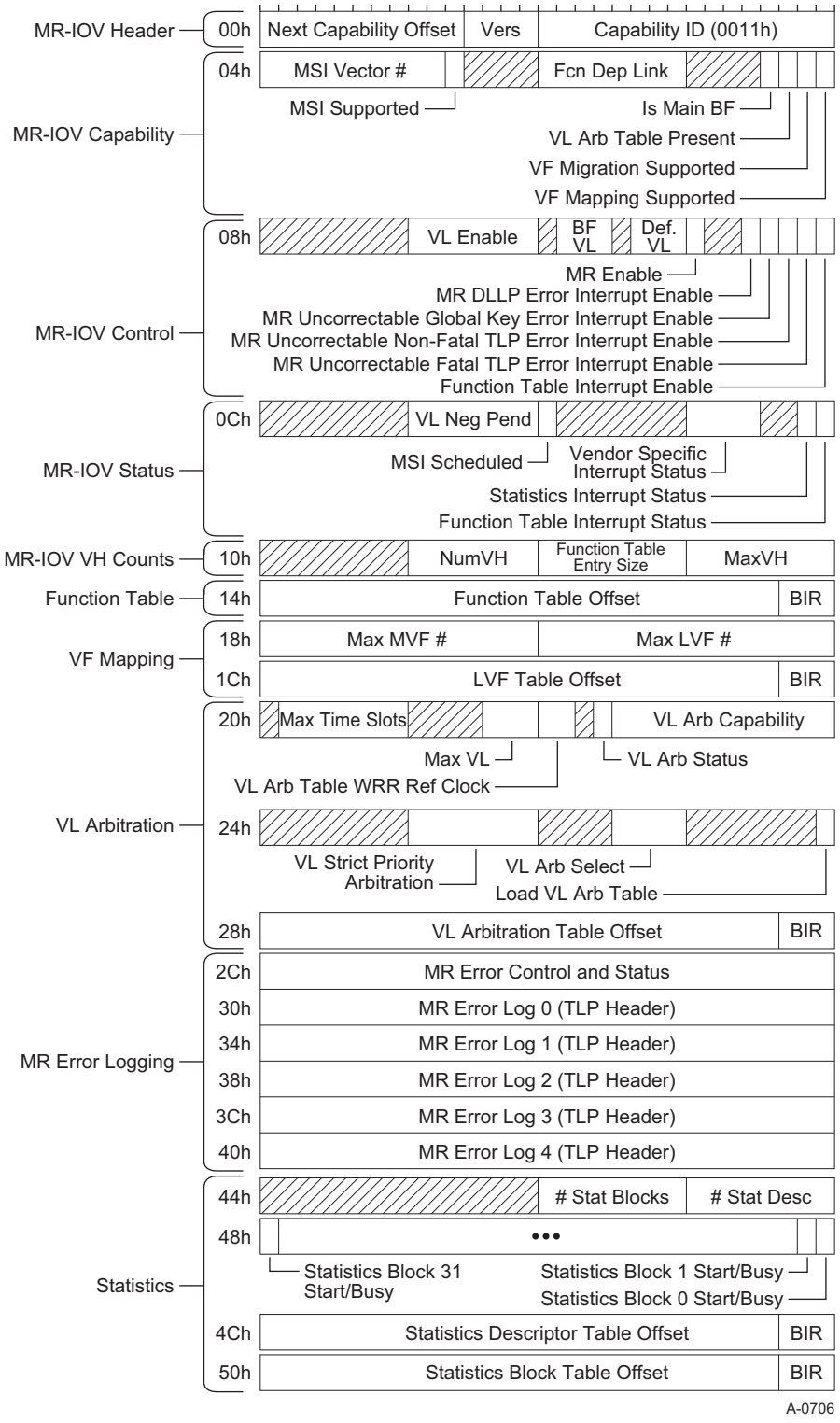

Figure 4-2: Device MR-IOV Capability

4.2.1.1. MR-IOV Extended Capability Header (00h)

Table 4-2 defines the Switch MR-IOV Extended Capability header. The Capability ID for the Switch MR-IOV Extended Capability is 0011h.

Table 4-2: Device MR-IOV Extended Capability Header

Bit Location	Register Description	Attributes
15:0	**PCI Express Extended Capability ID** – This field is a PCI-SIG defined ID number that indicates the nature and format of the Extended Capability. The Extended Capability ID for the MR-IOV Extended Capability is 0011h.	RO
19:16	**Capability Version** – This field is a PCI-SIG defined version number that indicates the version of the Capability structure present. Must be 1h for this version of the specification.	RO
31:20	**Next Capability Offset** – This field contains the offset to the next PCI Express Capability structure or 000h if no other items exist in the linked list of Capabilities. This offset is relative to the beginning of PCI compatible Configuration Space and thus must always be either 000h (for terminating list of Capabilities) or greater than 0FFh.	RO

4.2.1.2. MR-IOV Capabilities (04h)

Table 4-3: MR-IOV Capabilities

Bit Location	Register Description	Attributes
0	**VF Mapping Supported** – If Set, this BF supports mapping of Mission Virtual Functions (MVFs) into VFs. If Clear, any mapping is Vendor Specific and is not controlled by MR-PCIM.	RO
1	**VF Migration Supported** – If Set, this BF supports dynamic mapping of Mission Virtual Functions into and out of VFs. If Clear, VF Migration support is not available. If VF Migration Supported is Set, VF Mapping Supported must also be Set. VF Migration cannot occur unless enabled by software. VF Migration is enabled on a per-PF basis. See Section 3.2.4 for details.	RO
2	**VL Arbitration Table Present** – If Set, the VL Arbitration Table is present. If Clear, the VL Arbitration Table is not present and the corresponding fields in this BF are Read Only Zero. This field is only meaningful in the Main BF of the Device. In all other BFs, this field is Read Only Zero.	RO
3	**Is Main BF** – If Set, this BF is the "Main" BF of the Device. Certain fields are only meaningful in the Main BF.	RO
7:4	**Reserved**	RO
15:8	**Function Dependency Link** – The programming model for a Device may have Vendor Specific dependencies between sets of Functions. The Function Dependency Link field is used to inform MR-PCIM about these dependencies. This field describes dependencies between BFs. PF and VF dependencies are the same as the dependencies of their associated BFs. If a BF is independent from other BFs of a Device, this field shall contain the Function Number of the BF. If a BF is dependent on other BFs of a Device, this field shall contain the Function Number of the next BF in the same Function Dependency List. The last BF in a Function Dependency List shall contain the Function Number of the first BF in the Function Dependency List. For BFs in a Function Dependency List, MR-PCIM must allocate MVFs consistently. For these BFs, every VH should contain the exact same mapping of MVFs to VFs.	RO
19:16	**Reserved**	RO
20	**MSI Supported** – If Set, indicates that this BF can generate MSI interrupts.	RO

Bit Location	Register Description	Attributes
31:21	**MSI Vector Number** – This field indicates the MSI Vector number used to signal MR-IOV events within this BF. This value may change based on whether MSI or MSI-X is enabled. This field is Read Only Zero if MSI Supported is Clear.	RO

4.2.1.3. MR-IOV Control (08h)

Table 4-4: Device MR-IOV Control

Bit Location	Register Description	Attributes
0	**Function Table Interrupt Enable** – If Set, when any bit in the Function Interrupt Status Bitmap is Set, an interrupt is requested. If Clear, transitions of the Function Interrupt Status Bitmap do not request an interrupt. Default is 0b.	RW
1	**Statistics Interrupt Enable** – Enables delivery of Statistics Interrupts. This bit is not implemented and Read Only Zero if the Number of Statistics Blocks is 0 (see Section 4.5.1.1). The default value of this field is 0b.	RW
2	**ReservedP**	RO
3	**MR Uncorrectable Fatal TLP Error Interrupt Enable** – When both this bit and the MR Uncorrectable Fatal Error Status bit are Set, an interrupt is requested. Default is 0b. This bit is Read Only Zero if Is Main BF is Clear.	RW
4	**MR Uncorrectable Non-Fatal TLP Error Interrupt Enable** – When both this bit and the MR Uncorrectable Non-Fatal Error Status bit are Set, an interrupt is requested. Default is 0b. This bit is Read Only Zero if Is Main BF is Clear.	RW
5	**MR Uncorrectable Global Key Error Interrupt Enable** – When both this bit and the MR Uncorrectable Global Key Error Status bit are set, an interrupt is requested. Default is 0b. This bit is Read Only Zero if Is Main BF is Clear.	RW
6	**MR DLLP Error Interrupt Enable** – When both this bit and the MR DLLP Error Status bit are Set, an interrupt is requested. Default is 0b. This bit is Read Only Zero if Is Main BF is Clear.	RW
7	**MR Enable** – If Set, NumVH is Read Only and software may enable additional VLs and VHs. If Clear, NumVH may be written and additional VHs and VLs cannot be enabled. This field is only meaningful in the Main BF of the Device. In all other BFs, it is Read Only Zero. Default is 0b. Hardware behavior on the 1b to 0b transition of this field when LinkUp is 1b is undefined.	RW

Bit Location	Register Description	Attributes
10:9	**Default VL** – VL Number used for TLPs originated in VH0 and not associated with a BF, PF, or VF (i.e., a "Plain Old Function"). Such TLPs must use TC 0/VC 0. This field is only meaningful in the Main BF of the Device. In all other BFs, it is Read Only Zero. If all functions in VH0 are BFs, PFs, or VFs, this field may be Read Only Zero. If MaxVL is 0h, this field is Read Only Zero. Behavior is undefined if the VL Number in this field has VL Enable Clear or has VL Negotiation Pending Set or if the value in this field is greater than MaxVL.	RW
11	**ReservedP**	RO
14:12	**BF VL** – VL Number used for TLPs originating from a BF. Such TLPs must use TC 0/VC 0. This field is only meaningful in the Main BF of the Device. In all other BFs, it is Read Only Zero. If MaxVL is 0h, this field is Read Only Zero. Behavior is undefined if the VL Number in this field has VL Enable Clear or has VL Negotiation Pending Set or if the value in this field is greater than MaxVL.	RW
15	**ReservedP**	RO

Bit Location	Register Description	Attributes
23:16	**VL Enable** –This bit, when Set, enables a Virtual Link (see note 1 for exceptions). The Virtual Link is disabled when this bit is cleared. Software must use the VL Negotiation Pending bit to check whether the VL negotiation is complete. Bit 0 of this field is associated with VL0. Bit 1 of this field is associated with VL1, etc. The default value of this field is 00000001b (i.e. 1b for VL0 and is 0b for other VLs). The number of bits implemented in this field is determined by the MaxVL value. VL Enable bits for VLs greater than MaxVL are Read Only Zero. Behavior is undefined if VL Enable for non-zero VLs is Set when MR Enable is Clear. This field is only meaningful in the Main BF of the Device. In all other BFs, it is Read Only Zero. Notes: 1. This bit is hardwired to 1b for the VL0; i.e., writing to this bit has no effect for VL0. 2. To enable a Virtual Link, the VL Enable bits for that Virtual Link must be Set in both components on a Link. 3. To disable a Virtual Link, the VL Enable bits for that Virtual Link must be cleared in both components on a Link. 4. Software must ensure that no traffic is using a Virtual Link at the time it is disabled. 5. Software must fully disable a Virtual Link in both components on a Link before re-enabling the Virtual Link.	RW
31:24	**ReservedP**	RO

4.2.1.4. MR-IOV Status (0Ch)

Table 4-5: Device MR-IOV Status

Bit Location	Register Description	Attributes
0	**Function Table Interrupt Status** – Set if any bit in the Function Interrupt Status Bitmap is Set.	RO
1	**Statistics Interrupt Status** – This bit is Set when a Statistics Collection process completes. This bit is Read Only Zero if the Number of Statistics Blocks is zero (see Section 4.5.1.1)	RW1C
14:2	**ReservedZ**	RO
15	**MSI Scheduled** – Set when an MSI has been requested. If Set, subsequent MSIs are suppressed. If Clear, any enabled interrupt will cause an MSI to be scheduled (and this bit to be Set). Default is 0b. This field is Read Only Zero if MSI Supported is Clear.	RW1C
23:16	**VL Negotiation Pending** – These bits indicate whether Flow Control negotiation for a VL is in the pending state. The values of these bits are defined only when the Link is in the DL_Active state and the Virtual Link is enabled (its VL Enable bit is Set). When these bits are Set by hardware, it indicates that the VL resource has not completed the process of negotiation. This bit is cleared by hardware after the VL negotiation is complete (on exit from the MR FC_INIT2 state on the VL). This field is only meaningful in the Main BF of the Device. In all other BFs, this field is Read Only Zero.	RO
31:24	**ReservedZ**	RO

4.2.1.5. MR-IOV VH Counts (10h)

Table 4-6: Device MR-IOV VH Counts

Bit Location	Register Description	Attributes
7:0	**MaxVH** – Maximum number of VHs minus 1 supported on this Device. VHs are numbered [0..MaxVH]. This field is only meaningful in the Main BF of the Device. In all other BFs, this field is Read Only Zero.	RO
15:8	**Function Table Entry Size** – Returns the size, in DWORDs of each Function Table Entry. For this version of the specification, this value is at least 16.	RO
23:16	**NumVH** – Indicates the number of VHs enabled by the upstream component. This value must be less than or equal to MaxVH. The default value of this field is 0 indicating one VH is enabled. This field is only meaningful in the Main BF of the Device. In all other BFs, this field is Read Only Zero. This field is Read Only if MR Enable is Set. This field is Read/Write if MR Enable is Clear.	RW
31:24	**ReservedZ**	RO

4.2.1.6. Function Table Offset (14h)

Table 4-7: Device Function Table Offset

Bit Location	Register Description	Attributes
2:0	**Function Table BIR** – Indicates which one of a function's Base Address registers, located beginning at 10h in Configuration Space, is used to map the Function's Function Table into Memory Space. **BIR Value Base Address register** 0 BAR0 10h 1 BAR1 14h 2 BAR2 18h 3 BAR3 1Ch 4 BAR4 20h 5 BAR5 24h 6..7 Reserved For a 64-bit Base Address register, the BIR indicates the lower DWORD.	RO
31:3	**Function Table Offset** – Used as an offset from the address contained by one of the Function's Base Address registers to point to the base of the Function Table. The lower 3 BIR bits are masked off (set to zero) by software to form a 32-bit offset that is QWORD aligned. The minimum value of this field is 4 (the Function Interrupt Status Bitmap is 20h bytes and preceeds the Functino Table, see Section 4.2.4.7)	RO

The total size of the table (in bytes) is:

((MaxVH + 1) * Function_Table_Entry_Size * 4) + 32

This includes both the Function Table and the Function Interrupt Status Bitmap.

The Function Interrupt Status Bitmap immediately precedes the Function Table. The size of the Function Table Interrupt Bitmap is 32 bytes (supporting a maximum of 256 VHs). Bit 0 of the first DWORD corresponds to VH 0; bit 1 corresponds to VH 1, … Any unused bits are Read Only Zero.

4.2.1.7. MVF and LVF Sizes (18h)

If VF Mapping is supported, this register indicates the MVF Index values that may be assigned to a VF. Values in the range [1..Max MVF] may be written to the LVF Table mapping field.

Table 4-8: VF MVF Region

Bit Location	Register Description	Attributes
15:0	**Max LVF** – Indicates size of the LVF Table. The LVF Table contains MaxLVF+1 entries numbered 0 to Max LVF. If VF Mapping is not supported, this field is Zero. All BFs from the same Function Dependency Group must have the same Max LVF value.	RO
31:15	**Max MVF** – Indicates the highest MVF Index that can be assigned to a VF. Zero if VF Mapping Supported is Clear. MVF numbers are in the range [1..Max MVF]. If VF Mapping is not supported, this field is Zero. All BFs from the same Function Dependency Group must have the same Max MVF value.	RO

4.2.1.8. LVF Table Offset (1Ch)

Table 4-9: LVF Table Offset

Bit Location	Register Description	Attributes
2:0	**LVF Table BIR** – Indicates which one of a function's Base Address registers, located beginning at 10h in Configuration Space, is used to map the Function's LVF Table into Memory Space. **BIR Value** — **Base Address register** 0 — BAR0 10h 1 — BAR1 14h 2 — BAR2 18h 3 — BAR3 1Ch 4 — BAR4 20h 5 — BAR5 24h 6..7 — Reserved For a 64-bit Base Address register, the BIR indicates the lower DWORD.	RO
31:3	**LVF Table Offset** – Used as an offset from the address contained by one of the Function's Base Address registers to point to the base of the VH Table. The lower 3 BIR bits are masked off (set to zero) by software to form a 32-bit offset that is QWORD aligned.	RO

The total size of the LVF Table (in bytes) is: Max LVF * 4

4.2.1.9. VL Arbitration Capability and Status (20h)

Table 4-10: Device VL Arbitration Capability and Status

Bit Location	Register Description	Attributes
11:0	**VL Arbitration Capability** – Indicates the types of VL Arbitration supported by the Port. This field is valid for all Functions that report a Low Priority Extended VC Count field greater than 0. For all other Functions, this field must be hardwired to 00h. Each bit location within this field corresponds to a VC Arbitration Capability defined below. When more than 1 bit in this field is Set, it indicates that the Port can be configured to provide different VC arbitration services. Defined bit positions are: Bit 0 Hardware fixed arbitration scheme, e.g., Round Robin Bit 1 Weighted Round Robin (WRR) arbitration with 32 phases Bit 2 WRR arbitration with 64 phases Bit 3 WRR arbitration with 128 phases Bit 4 Time-based WRR with 128 phases Bit 5 WRR Arbitration with 256 phases Bits 6-7 Reserved Bit 10-8 Vendor Defined VL Arbitration Scheme This field is Read Only Zero if VL Arbitration Present is Clear.	RO
12	**VL Arb Status** – This bit indicates the coherency status of the VL Arbitration Table. This bit is valid only when the VL Arbitration Table is used. This bit is Set by hardware when any entry of the VL Arbitration Table is written to by software. This bit is cleared by hardware when hardware finishes loading values stored in the VL Arbitration Table after software sets the Load VL Arbitration Table bit. Default value of this bit is 0b. This field is Read Only Zero if VL Arbitration Present is Clear.	RO
13	**Reserved**	RO
15:14	**Reference Clock** – Indicates the reference clock for Virtual Links that support time-based WRR VL Arbitration. This field is valid only if time-based WRR is supported. Defined encodings are: 00b 100 ns reference clock 01b – 11b Reserved This field is Read Only Zero if VL Arbitration Present is Clear.	RO
18:16	**MaxVL** – Indicates the number of VLs supported. The Device supports VL_0 through VL_{MaxVL} inclusive. This field is only meaningful in the Main BF of the Device. In all other BFs, it is Read Only Zero.	RO
23:19	**Reserved**	RO

Bit Location	Register Description	Attributes
30:24	**Maximum Time Slots** – Indicates the maximum number of time slots (minus one) that are supported when configured for time-based WRR VL Arbitration. For example, a value 000 0000b in this field indicates the supported maximum number of time slots is 1 and a value of 111 1111b indicates the supported maximum number of time slot is 128. This field is valid only when the VL Arbitration Capability field indicates that time-based WRR VL Arbitration is supported. This field is Read Only Zero if VL Arbitration Present is Clear.	RO
31	**Reserved**	RO

4.2.1.10. VL Arbitration Control (24h)

Table 4-11: Device VL Arbitration Control

Bit Location	Register Description	Attributes
0	**Load VL Arbitration Table** – When Set, this bit updates the VL Arbitration logic from the VL Arbitration Table. This bit is valid only when the VL Arbitration Table is used by the selected VL Arbitration scheme (that is indicated by a Set bit in the VL Arbitration Capability field selected by VL Arbitration Select). Software sets this bit to signal hardware to update VL Arbitration logic with new values stored in VL Arbitration Table; clearing this bit has no effect. Software uses the VL Arbitration Table Status bit to confirm whether the new values of VL Arbitration Table are completely latched by the arbitration logic. This bit always returns 0b when read. Default value of this bit is 0b. This field is Read Only Zero if VL Arbitration Present is Clear.	RW
7:4	**Reserved**	RO
11:8	**VL Arbitration Select** – This field configures the Port to provide a particular VL Arbitration service. The permissible value of this field is a number corresponding to one of the asserted bits in the VL Arbitration Capability field. This field is Read Only Zero if VL Arbitration Present is Clear.	RW
15:12	**Reserved**	RO

Bit Location	Register Description	Attributes
23:16	**VL Strict Priority Arbitration** – This field contains one bit per VL. Bit 0 corresponds to VL0. Bit 7 corresponds to VL7. When a bit is Set, the corresponding VL is configured to arbitrate as Strict Priority based on VL number. When a bit is Clear, the corresponding VL is configured to arbitrate as normal priority (using the scheme selected by VL Arbitration Select). Among the VLs configured for strict priority, priority is based on increasing VL number. VL0 is the lowest strict priority; VL7 is the highest. Strict Priority VLs have priority over normal priority VLs. Behavior is Undefined if a VL configured for Strict Priority is also included in the VL Arbitration Table. If a VL is Disabled, the value of the corresponding bit in this field is ignored. Default value of this field is 0000 0000b. This field is Read Only Zero if VL Arbitration Present is Clear.	RW
31:24	**Reserved**	RO

4.2.1.11. VL Arbitration Table Offset (28h)

Table 4-12: Device VL Arbitration Table Offset

Bit Location	Register Description	Attributes
2:0	**VL Arbitration Table BIR** – Indicates which one of a function's Base Address registers, located beginning at 10h in Configuration Space, is used to map the Function's VL Arbitration Table into Memory Space. **BIR Value Base Address register** 0 BAR0 10h 1 BAR1 14h 2 BAR2 18h 3 BAR3 1Ch 4 BAR4 20h 5 BAR5 24h 6..7 Reserved For a 64-bit Base Address register, the BIR indicates the lower DWORD. This field is Read Only Zero if VL Arbitration Present is Clear.	RO
31:3	**VL Arbitration Table Offset** – Used as an offset from the address contained by one of the Function's Base Address registers to point to the base of the VL Arbitration Table. The lower 3 BIR bits are masked off (set to zero) by software to form a 32-bit offset that is QWORD aligned. This field is Read Only Zero if VL Arbitration Present is Clear.	RO

4.2.1.12. MR Error Status (2Ch)

This register is Read Only Zero if Is Main BF is Clear.

Table 4-13: Device MR Error Status

Bit Location	Register Description	Attributes
3:0	**MR First Error Pointer** – The First Error Pointer is a field that identifies the bit position of the first error reported in the MR Error Status register. Default is 0h.	RO
4	**MR Uncorrectable Fatal TLP Error Status** – This bit is Set when the Uncorrectable Fatal TLP Error Status is Set and the Uncorrectable Fatal TLP Error Mask bit is Clear. Default is 0b.	RW1C
5	**MR Uncorrectable Non-Fatal TLP Error Status** – This bit is Set when the Uncorrectable Non-Fatal TLP Error Status is Set and the Uncorrectable Non-Fatal TLP Error Mask bit is Clear. Default is 0b.	RW1C
6	**MR Uncorrectable Global Key Error Status** – This bit is Set when the Uncorrectable Global Key Error Status is Set and the Uncorrectable Global Key Error Mask bit is Clear. Default is 0b.	RW1C
7	**Reserved**	RsvdZ
8	**MR DLLP Error Status** – This bit is Set when the Uncorrectable Fatal TLP Error Status is Set and the Uncorrectable Fatal TLP Error Mask bit is Clear. Default is 0b. Headers are not logged and the First Error Pointer is not updated for DLLP Errors.	RW1C
14:9	**Reserved**	RsvdZ
15	**MR Multiple Uncorrectable Error** – Set when hardware detects an MR Uncorrectable Error but is unable to indicate it because the Error Status bit is already Set. Default is 0b.	RW1C

4.2.1.13. MR Error Control (2Eh)

This register is Read Only Zero if Is Main BF is Clear.

Table 4-14: Device MR Error Control

Bit Location	Register Description	Attributes
3:0	**Reserved**	RO
4	**MR Uncorrectable Fatal TLP Error Mask** – If this bit is Set, Uncorrectable Fatal TLP Errors are not logged and the MR Uncorrectable Fatal TLP Error Status will never be Set. Default is 0b.	RW
5	**MR Uncorrectable Non-Fatal TLP Error Mask** – If this bit is Set, Uncorrectable Non-Fatal TLP Errors are not logged and the MR Uncorrectable Non-Fatal TLP Error Status will never be Set. Default is 0b.	RW
6	**MR Uncorrectable Global Key Error Mask** – If this bit is Set, Uncorrectable Global Key Errors are not logged and the MR Uncorrectable Global Key Error Status will never be Set. Default is 0b.	RW
7	**Reserved**	RsvdP
8	**MR DLLP Error Mask** – If this bit is Set, Uncorrectable Fatal TLP Errors are not logged and the MR Uncorrectable Fatal TLP Error Status will never be Set. Default is 0b.	RW
15:9	**Reserved**	RsvdP

4.2.1.14. MR Header Log (30h to 40h)

These fields contain the TLP Prefix and TLP header corresponding to the error described by the First Error Pointer in the MR Error Status register.

The value of these fields is undefined if the First Error Pointer is zero or points to a bit number that is not Set.

Headers are not logged and the First Error Pointer is not updated for DLLP Errors.

4.2.1.15. Statistics Capability and Control (44h to 50h)

Device and Switch Statistics related fields are described in Section 4.5.

4.2.2. Device VL Arbitration Table

Switch and Device VL Arbitration tables are identical. See Section 4.3.7.3 for details.

4.2.3. LVF Table

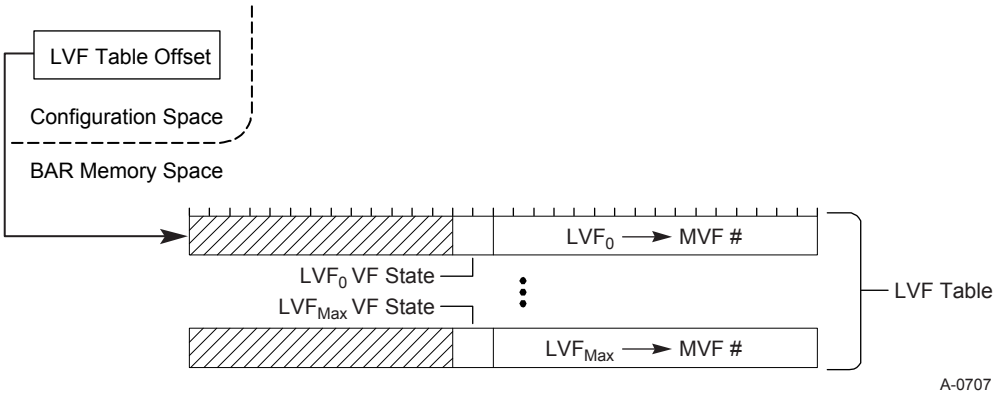

Figure 4-3: LVF Table

The LVF Table is present if VF Mapping Supported is Set.

LVF Table Entries are one DWORD.

4.2.3.1. LVF Table Entry

Table 4-15: LVF Table Entry

Bit Location	Register Description	Attributes
15:0	**MVF # for this LVF** – MVF mapped to this LVF. Only enough bits are implemented to express values in the range [0..Max MVF]. Unimplemented high order bits are read only zero. The value 0 indicates this LVF is not mapped to any MVF. Default value is Vendor Specific.	RW
17:16	**VF State** – VF Migration State for this LVF. Values are: 00 Inactive.Unavailable 01 Inactive.MigrateIn 11 Active.Available 10 Active.MigrateOut MR-PCIM changes VF Migration state by writing this value. If VF Migration is not supported, this field is Read Only Zero. Even when VF Migration is Disabled, MR-PCIM must ensure that this field is meaningful (i.e., VFs below InitialVFs should be Active.Available).	RW
31:18	**Reserved**	RO

4.2.4. Function Table

The Function table contains one entry for each VH. The first entry represents VH0; the second entry represents VH1, etc.

Every non-zero VH contains exactly one PF or Non-IOV Function associated with this BF. VH0 can optionally have a single PF or Non-IOV Function. A BF may not have a mixture of PFs and Non-IOV Functions associated with it (i.e., if a BF is associated with a PF in one VH, then it may not be associated with a Non-IOV Function in another VH).

Certain Function Table Entry fields are only implemented in the Main BF of the Device (for example, the Global Key fields and MFVC related fields).

The first Function Table Entry is always present. If VH0 does not contain a PF, most of the fields are Read Only Zero but a few fields remain meaningful (e.g., Global Key).

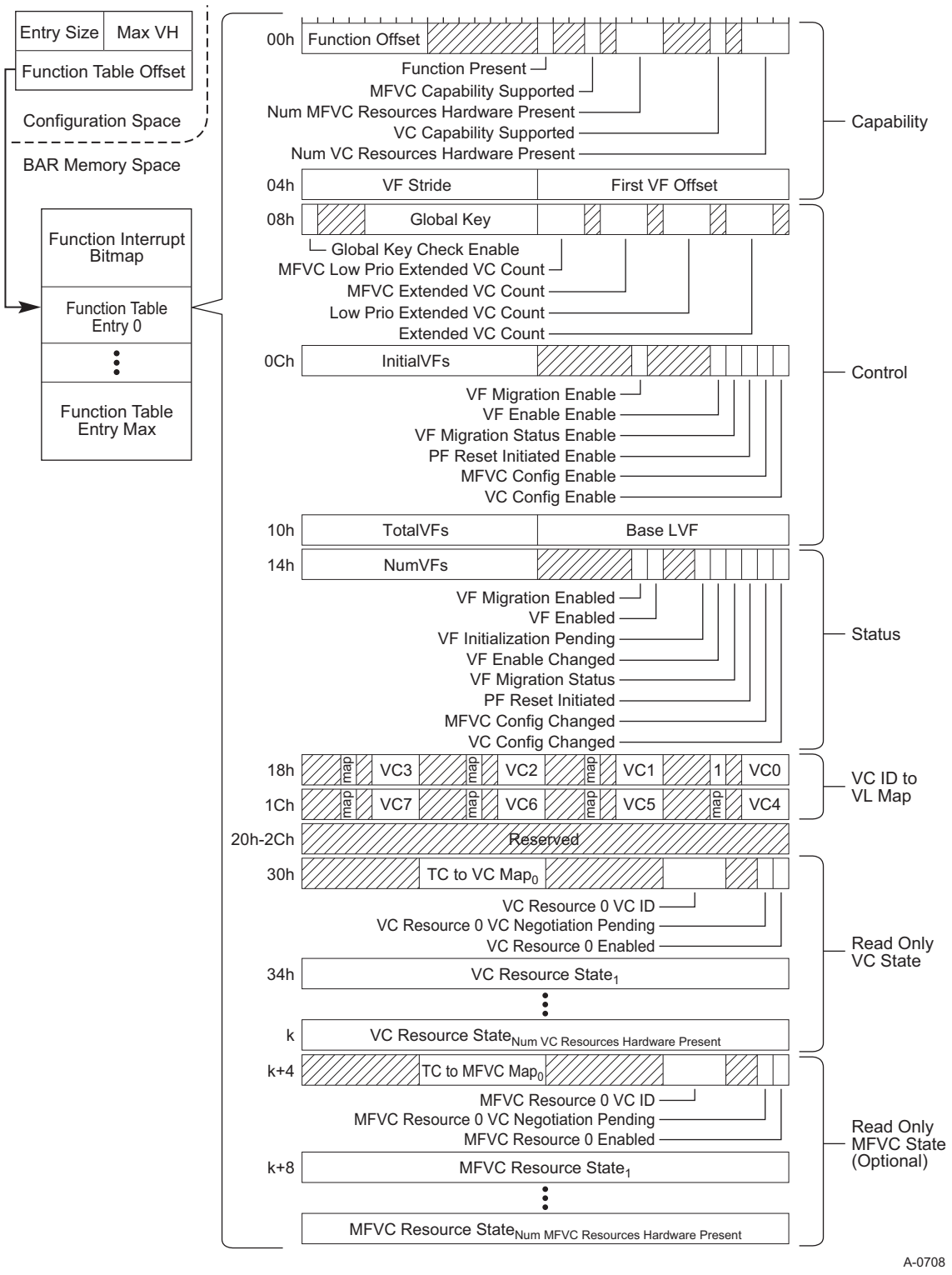

Figure 4-4: Device Function Table

4.2.4.1. Function Capability (00h and 04h)

Table 4-16: Function Capability 1 (00h)

Bit Location	Register Description	Attributes
2:0	**Num VC Resources Hardware Present** – Indicates the index of the last VC Resource array structure implemented in this Function's VC Capability. The value 0 indicates one VC Resource is provided. The value 7 indicates that all 8 VC Resources are provided. This value indicates the number that the hardware implements. MR-PCIM software may offer a lower number to the VH by setting VC Extended VC Count. If VC Capability Supported is Clear, this field is hardwired to 000b.	RO
3	**Reserved**	RO
4	**VC Capability Supported** – If Set, the Function contains a VC Capability. If Clear, the Function does not contain a VC Capability.	RO
7:5	**Reserved**	RO
10:8	**Num MFVC Resources Hardware Present** – Indicates the index of the last MFVC Resource array structure implemented in this Function's MFVC Capability. The value 0 indicates one MFVC Resource is provided. The value 7 indicates that all 8 MFVC Resources are provided. This value indicates the number that the hardware implements. MR-PCIM software may offer a lower number to the VH by setting MFVC Extended VC Count. If MFVC Capability Supported is Clear, this field is hardwired to 000b.	RO
11	**Reserved**	RO
12	**MFVC Capability Supported** – If Set, in every VH, the PF associated with this BF contains an MFVC Capability. If Clear, the PF associated with this BF does not contain an MFVC Capability. This bit may only be Set in the BF that manages Function 0 within the VH (i.e., Function Offset is 00h).	RO
15:13	**Reserved**	RO
16	**Function Present** – If Set, indicates this BF has a Function in VH0. If Clear, this BF does not have a Function in VH0. This field is only meaningful for the first Function Table Entry (i.e., VH0) and is reserved in all other Function Table Entries.	RO/ Reserved
23:17	**Reserved**	RO

Bit Location	Register Description	Attributes
31:24	**Function Offset** – Contains the low 8 bits of the Function's RID within the associated VH. If the Captured Bus Number is NNh, the Function's RID is Function Offset plus NN00h. For the first Function Table Entry (i.e., VH0), Function Offset is meaningful only if Function Present is Set. For the remaining Function Table Entries, Function Offset is always meaningful. With the exception of the first Function Table Entry (i.e., VH0), all Function Offset values for a given BF must have the same value.	RO

Table 4-17: Function Capability 2 (04h)

Bit Location	Register Description	Attributes
15:0	**First VF Offset** – Contains the First VF Offset of this PF's SR-IOV Capability. Zero if this Function is not a PF (i.e., it has no SR-IOV Capability).	RO
31:16	**VF Stride** – Contains the VF Stride of this PF's SR-IOV Capability. Zero if this Function is not a PF (i.e., it has no SR-IOV Capability).	RO

4.2.4.2. Function Control (08h to 10h)

Table 4-18: Function Control 1 (08h)

Bit Location	Register Description	Attributes
0	**Reserved**	RO
3:1	**VC Extended VC Count** – The value presented to software in the VH in the Extended VC Count field of the VC Capability. Valid values are [0..Num VC Hardware Resources Present]. MR-PCIM may set this value to restrict the number of VCs offered to a VH. Default value is 0h. If VC Capability Supported is Clear, this field is hardwired to 0h.	RW
4	**Reserved**	RO
7:5	**VC Low Priority Extended VC Count** – The value presented to software in the VH in the Low Priority Extended VC Count field of the VC Capability. Valid values are [0..VC Extended VC Count]. The value of this field does not affect arbitration in any manner. This field allows MR-PCIM to indicate to software in the VH which VCs it should think are strict priority arbitration. Default value is 0h. If VC Capability Supported is Clear, this field is hardwired to 0h.	RW
8	**Reserved**	RO
11:9	**MFVC Extended VC Count** – The value presented to software in the VH in the Extended VC Count field of the MFVC Capability. Valid values are [0..Num MFVC Hardware Resources Present]. MR-PCIM may set this value to restrict the number of VCs offered to a VH. Default value is 000b. If MFVC Capability Supported is Clear, this field is hardwired to 000b.	RW
12	**Reserved**	RO
15:13	**MFVC Low Priority Extended VC Count** – The value presented to software in the VH in the Low Priority Extended VC Count field of the MFVC Capability. Valid values are [0..MFVC Extended VC Count]. The value of this field does not affect arbitration in any manner. This field allows MR-PCIM to indicate to software in the VH which VCs it should think are strict priority arbitration. Default value is 000b. If MFVC Capability Supported is Clear, this field is hardwired to 000b.	RW

Bit Location	Register Description	Attributes
27:16	**Global Key** – TLPs received are checked against this value. TLPs sent contain this value. If this field contains 000h, the "wild card" value, checking is disabled. If a TLP contains 000h, the TLP is a wild card TLP and checking always passes. Default value of this field is 000h. This field is implemented only in the Main BF of the Device (i.e., Is Main BF is Set). This field is implemented in all Function Table entries of the Main BF even if Function Present is Clear. In BFs other than the Main BF, this field is hardwired to 000h.	RW
30:28	**Reserved**	RO
31	**Global Key Check Enable** – If Set, Global Key checking is performed. If Clear, Global Key mismatches are ignored. This field is implemented in all Function Table entries of the Main BF, even if Function Present is Clear. In BFs other than the Main BF, this field is hardwired to 0b.	RW

Table 4-19: Function Control 2 (0Ch)

Bit Location	Register Description	Attributes
0	**VC Config Interrupt Enable** – If Set, setting the VC Config Changed bit for this Function Table Entry sets the corresponding Function Interrupt Status Bitmap. If Clear, the Function Interrupt Status Bitmap associated with this Function Table Entry is not affected by VC Config Changed. Default is 0b. This field is Read Only Zero if VC Capability Supported is Clear.	RW
1	**MFVC Config Changed Enable** – If Set, setting the MFVC Config Changed bit for this Function Table Entry sets the corresponding Function Interrupt Status Bitmap. If Clear, the Function Interrupt Status Bitmap associated with this Function Table Entry is not affected by MFVC Config Changed. Default is 0b. This field is Read Only Zero if MFVC Capability Supported is Clear.	RW
2	**PF Reset Initiated Enable** – If Set, setting the PF Reset Initiated bit for this Function Table Entry sets the corresponding Function Interrupt Status Bitmap. If Clear, the Function Interrupt Status Bitmap associated with this Function Table Entry is not affected by PF Reset Initiated. Default is 0b. This field is Read Only Zero if VF Migration Supported is Clear.	RW

Bit Location	Register Description	Attributes
3	**VF Migration Status Enable** – If Set, setting the VF Migration Status bit for this Function Table Entry sets the corresponding Function Interrupt Status Bitmap. If Clear, the Function Interrupt Status Bitmap associated with this Function Table Entry is not affected by VF Migration State. Default is 0b. This field is Read Only Zero if VF Migration Supported is Clear.	RW
4	**VF Enable Enable** – If Set, setting the VF Enable changed bit for this Function Table Entry sets the corresponding Function Interrupt Status Bitmap. If Clear, the Function Interrupt Status Bitmap associated with this Function Table Entry is not affected by VF Enable Changed. Default is 0b. This field is Read Only Zero if VF Mapping Supported is Clear.	RW
8:5	**Reserved**	RO
9	**VF Migration Capable** – Set by MR-PCIM to indicate to SR-PCIM that VF Migration support is available. VF Migration is possible only if this bit and VF Migration Enabled are both set. Default is 0b. This field is Read Only Zero if VF Migration Supported is Clear.	RW
15:10	**Reserved**	RO
31:16	**InitialVFs** – Number of VFs provided to SR-PCIM and populated by MR-PCIM with MVFs. This field is meaningful only for PFs. If this Function does not contain an SR-IOV Capability, this field is Read Only Zero. If VF Mapping is not supported on this PF, this field is Read Only and indicates the number of VFs that were provisioned using Vendor Specific mechanisms. If VF Mapping is supported, the default value of this field is 0000h. The value 0 indicates that no populated VFs are offered to SR-PCIM.	RW

Table 4-20: Function Control 3 (10h)

Bit Location	Register Description	Attributes
15:0	**Base LVF** – Set by MR-PCIM to contain the index of first LVF table entry assigned to this PF. If VF Mapping is not supported, this field is Read Only Zero. The default value of this field is Vendor Specific.	RW
31:16	**TotalVFs** – Total Number of VFs provided to SR-PCIM including populated and non-populated VFs. This field is meaningful only if VF Migration is supported. If VF Migration is not supported, this field is Read Only Zero and the InitialVFs value is returned as TotalVFs to SR-PCIM. The default value of this field is 0000h. MR-PCIM must configure this field to be greater than or equal to InitialVFs. The value 0 indicates that no VFs are offered to SR-PCIM.	RW

4.2.4.3. Function Status (14h)

Setting of any of bits 7:0 of this register Sets the corresponding Function Interrupt Status Bitmap entry if the Function Interrupt Enable bit is Set. When software clears all of these bits, or clears the Function interrupt Enable bit, the corresponding Function Interrupt Status Bitmap entry is also cleared.

Table 4-21: Function Status

Bit Location	Register Description	Attributes
0	**VC Config Changed** – Set when software in a VH changes any of the VC Resource State bits. Default is 0b. This field is Read Only Zero if VC Capability Supported is Clear.	RW1C
1	**MFVC Config Changed** – Set when software in a VH changes any of the MFVC Resource State bits. Default is 0b. This field is Read Only Zero if MFVC Capability Supported is Clear.	RW1C
2	**PF Reset Initiated** – Set when the PF is Reset. This can be due to a Reset DLLP within the VH or due to software in the VH issuing a Function Level Reset (FLR) to the PF. Default is 0b. This field is Read Only Zero if VF Migration Supported is Clear. Note: This bit provides MR-PCIM an indication that a reset has occurred so it can configure VF Migration State to a valid VF Migration Initial State (see Section 3.2.4.2). This bit is not needed in other situations.	RW1C
3	**VF Migration Status** – Set when a VF Migration event is triggered for some VF associated with this PF. See Section 3.2.4 for details. Default is 0b. This field is Read Only Zero if VF Migration Supported is Clear.	RW1C
4	**VF Enable Changed** – Set when SR-PCIM changes VF Enable. This field is Read Only Zero if VF Mapping Supported is Clear.	RW1C
5	**VF Initialization Pending** – Set when VF Migration Capable is Set and either the PF is Reset or VF Enable is Cleared. When Set, access within the VH to VF Configuration Space will return CRS. Default is 0b. This field is Read Only Zero if VF Migration Supported is Clear. MR-PCIM software should reestablish a valid initial VF Configuration and Clear this bit within 1 second. See Section 3.2.4.2 for details.	RW1C
7:6	**Reserved**	RO
8	**VF Enabled** – MR-IOV visible copy of the SR-IOV VF Enable bit.	RO
9	**VF Migration Enabled** – MR-IOV visible copy of the SR-IOV VF Migration Enable bit. This field is Read Only Zero if VF Migration Supported is Clear.	RO
15:10	**Reserved**	RO
31:16	**NumVFs** – MR-IOV visible copy of the SR-IOV NumVFs value.	RO

4.2.4.4. Function VC to VL Map (18h and 1Ch)

This table maps VCs in a VH to VLs. To preserve PCI Express ordering and flow control independence assumptions, each VC must be assigned to a distinct VL.

This map is only meaningful on the Main BF of the Device. In all other BFs, all fields of this map are Read Only Zero.

These fields contain the Virtual Link to be used for traffic out of this Function for the indicated VC. VC to VL mapping is not needed and these fields are read only zero if the MaxVL value is zero.

Table 4-22: Function Table VC to VL Map 1 (VC Capability)

Bit Location	Register Description	Attributes
2:0	**VC0 VL Map** – Indicates the VL number used for traffic labeled VC0. If VC Capability Supported is Clear or if MaxVL is 0h, this field is Read Only Zero. This field is Read Only when VC0 VL Map Enable is Set. The default value of this field is 0h.	RW
3	**Reserved**	RO
4	**VC0 VL Map Enable** – Indicates that the VC0 VL Map field contains a valid VL number. If VC Capability Supported is Clear or if MaxVL is 0h, this field is hardwired to 1b (VH0) or 0b (all other VHs). The default value of this field is 1b for VH0 and 0b for all other VHs. Hardware behavior on the 1b to 0b transition of this bit is undefined. Hardware behavior is undefined if MR Enable is Clear and this bit is Set on any non-zero VH.	RW
7:5	**Reserved**	RO
10:8	**VC1 VL Map** – Indicates the VL number used by traffic labeled VC1. This field is Read Only Zero if Num VC Resources Hardware Present is 0. This field is Read Only when VC1 VL Map Enable is Set. The default value of this field is 1h.	RW
11	**Reserved**	RO
12	**VC1 VL Map Enable** – Indicates that the VC1 VL Map field contains a valid VL number. This field is Read Only Zero if Num VC Resources Hardware Present is 0. The default value of this field is 0b. Hardware behavior on the 1b to 0b transition of this bit is undefined. Hardware behavior is undefined if MR Enable is Clear and this bit is Set.	RW
15:13	**Reserved**	RO

Bit Location	Register Description	Attributes
18:16	**VC2 VL Map** – Indicates the VL number used by traffic labeled VC2. This field is Read Only Zero if Num VC Resources Hardware Present is 0 or 1. This field is Read Only when VC2 VL Map Enable is Set. The default value of this field is 2h.	RW
19	**Reserved**	RO
20	**VC2 VL Map Enable** – Indicates that the VC2 VL Map field contains a valid VL number. This field is Read Only Zero if Num VC Resources Hardware Present is 0 or 1. The default value of this field is 0b. Hardware behavior on the 1b to 0b transition of this bit is undefined. Hardware behavior is undefined if MR Enable is Clear and this bit is Set.	RW
23:21	**Reserved**	RO
26:24	**VC3 VL Map** – Indicates the VL number used by traffic labeled VC3. This field is Read Only Zero if Num VC Resources Hardware Present is less than or equal to 2. This field is Read Only when VC3 VL Map Enable is Set. The default value of this field is 3h.	RW
27	**Reserved**	RO
28	**VC3 VL Map Enable** – Indicates that the VC3 VL Map field contains a valid VL number. This field is Read Only Zero if Num VC Resources Hardware Present is less than or equal to 2. The default value of this field is 0b. Hardware behavior on the 1b to 0b transition of this bit is undefined. Hardware behavior is undefined if MR Enable is Clear and this bit is Set.	RW
31:29	**Reserved**	RO

Table 4-23: Function Table VC to VL Map 2 (VC Capability)

Bit Location	Register Description	Attributes
3:0	**VC4 VL Map** – Indicates the VL number used by traffic labeled VC4. This field is Read Only Zero if Num VC Resources Hardware Present is less than or equal to 3. This field is Read Only when VC4 VL Map Enable is Set. The default value of this field is 4h.	RW
4	**Reserved**	RO
5	**VC4 VL Map Enable** – Indicates that the VC4 VL Map field contains a valid VL number. This field is Read Only Zero if Num VC Resources Hardware Present is less than or equal to 3. The default value of this field is 0b. Hardware behavior on the 1b to 0b transition of this bit is undefined. Hardware behavior is undefined if MR Enable is Clear and this bit is Set.	RW
7:6	**Reserved**	RO
10:8	**VC5 VL Map** – Indicates the VL number used by traffic labeled VC5. This field is Read Only Zero if Num VC Resources Hardware Present is less than or equal to 4. This field is Read Only when VC5 VL Map Enable is Set. The default value of this field is 5h.	RW
11	**Reserved**	RO
12	**VC5 VL Map Enable** – Indicates that the VC5 VL Map field contains a valid VL number. This field is Read Only Zero if Num VC Resources Hardware Present is less than or equal to 4. The default value of this field is 0b. Hardware behavior on the 1b to 0b transition of this bit is undefined. Hardware behavior is undefined if MR Enable is Clear and this bit is Set.	RW
15:13	**Reserved**	RO
18:16	**VC6 VL Map** – Indicates the VL number used by traffic labeled VC6. This field is Read Only Zero if Num VC Resources Hardware Present is less than or equal to 5. This field is Read Only when VC6 VL Map Enable is Set. The default value of this field is 6h.	RW
19	**Reserved**	RO
20	**VC6 VL Map Enable** – Indicates that the VC6 VL Map field contains a valid VL number. This field is Read Only Zero if Num VC Resources Hardware Present is less than or equal to 5. The default value of this field is 0b. Hardware behavior on the 1b to 0b transition of this bit is undefined. Hardware behavior is undefined if MR Enable is Clear and this bit is Set.	RW
23:21	**Reserved**	RO

Bit Location	Register Description	Attributes
26:24	**VC7 VL Map** – Indicates the VL number used by traffic labeled VC7. This field is Read Only Zero if Num VC Resources Hardware Present is less than or equal to 6. This field is Read Only when VC7 VL Map Enable is Set. The default value of this field is 7h.	RW
27	**Reserved**	RO
28	**VC7 VL Map Enable** – Indicates that the VC7 VL Map field contains a valid VL number. This field is Read Only Zero if Num VC Resources Hardware Present is less than or equal to 6. The default value of this field is 0b. Hardware behavior on the 1b to 0b transition of this bit is undefined. Hardware behavior is undefined if MR Enable is Clear and this bit is Set.	RW
31:29	**Reserved**	RO

4.2.4.5. Function VC Resource (30h up to 4Ch)

These fields return data from the VC Capability of the associated Type 0 Configuration header. They allow MR-PCIM software to track the enabling and mapping of VCs with each VH.

VC Resource State 0 is located at offset 30h; VC Resource State 1 is located at offset 34h; etc. Fields within VC Resource State are described in Table 4-24.

VC Resource State 0 always exists, even if VC Capability Supported is Clear.

VC Resource fields for resource numbers above VC Extended VC Count are hardwired to 0. VC Resource State fields for resource numbers above Num VC Resource Hardware Present are not implemented and do not exist.

Table 4-24: Function Table VC Resource State

Bit Location	Register Description	Attributes
0	**VC Enabled** – This field tracks the VC Enabled bit set by software operating in the VH. Per the *PCI Express Base Specification*, VC Resource 0 is always enabled and thus VC Enabled for VC Resource 0 is always set. If VC Capability Supported is Clear, this field is Set.	RO
1	**VC Negotiation Pending** – If VC Capability Supported is Set, this field tracks the VC Negotiation Pending bit view in the VH. It is Set when VC Enabled is Set and either no VL has been mapped to this VC (associated VL Map Enable bit is Clear) or Flow Control negotiation has not completed on the mapped VH and VL. If VC Capability Supported is Clear, this field is Set when either no VL has been mapped to this VC (associated VL Map Enable bit is Clear) or Flow Control negotiation has not completed on the mapped VH and VL.	RO
3:2	**Reserved**	RO
6:4	**VC ID** – This field tracks the VC ID field set by software operating in the VH. Per the *PCI Express Base Specification*, VC ID for VC Resource 0 is always 0. If VC Capability Supported is Clear, this field is 0.	RO
15:7	**Reserved**	RO
23:16	**TC to VC Map** – This field tracks the TC to VC Map field set by software operating in the VH. Per the *PCI Express Base Specification*, bit 0 of this field is fixed and the remaining bits may be set by software. Also, per the PCI Express Specification, the default value of this field is FFh for the first VC Resource and is 00h for other VC Resources. If VC Capability Supported is Clear, this field is FFh.	RO
31:24	**Reserved**	RO

4.2.4.6. Function Table MFVC Resource Status

These fields return data from the MFVC Capability of the associated Type 0 Configuration header. They allow MR-PCIM software to track the enabling and mapping of VCs with each VH.

The MFVC Resource fields are implemented only if MFVC Capability Supported is Set.

MFVC Resource fields for resource numbers above Extended MFVC Count are hardwired to 0. MFVC Resource fields for resource numbers above Num MFVC Resource hardware Present are not implemented and do not exist.

MFVC Resource State 0 is located immediately following the last Function VC Resource Status entry; MFVC Resource 1 is located immediately after MFVC Resource State 0; etc. If the VC Capability field is Clear indicating that only one Function VC Resource is present, MFVC Resource State 0 is located at offset 34h. Fields within MFVC Resource State are described in Table 4-25.

Table 4-25: VH Table MFVC Resource State

Bit Location	Register Description	Attributes
0	**VC Enabled** – This field tracks the VC Enabled bit set by software operating in the VH. Per the *PCI Express Base Specification*, MFVC Resource 0 is always enabled and thus VC Enabled for MFVC Resource 0 is always Set.	RO
1	**VC Negotiation Pending** – This field tracks the VC Negotiation Pending bit view in the VH. This bit is Set when VC Enabled is Set and either no VL has been mapped to this VC (associated VL Map Enable bit is Clear) or Flow Control negotiation has not completed on the mapped VH and VL.	RO
3:2	**Reserved**	RO
6:4	**VC ID** – This field tracks the VC ID field set by software operating in the VH. Per the *PCI Express Base Specification*, VC ID for MFVC Resource 0 is always 0.	RO
15:7	**Reserved**	RO
23:16	**TC to VC Map** – This field tracks the TC to VC Map field set by software operating in the VH. Per the *PCI Express Base Specification*, bit 0 of this field is fixed and the remaining bits may be set by software. Also, per the *PCI Express Base Specification*, the default value of this field is FFh for the first MFVC Resource and is 00h for other MFVC Resources.	RO
31:24	**Reserved**	RO

4.2.4.7. Function Interrupt Status Bitmap (minus 20h)

The Function Interrupt Status Bitmap precedes the Function Table. It is always 32 bytes (supporting a maximum of 256 Functions for each BF).

Bits in this table are Read Only. A bit is Set to indicate the Function has an interrupt pending and Clear otherwise. These Interrupt Status bits are cleared either by clearing the appropriate Function Interrupt Pending bit or by masking the interrupt using the Function Interrupt Enable.

An MSI Interrupt is requested on any zero to one transition of any of these bits.

Bits in the Function Interrupt Status bitmap are Read Only Zero if the associated Function cannot generate an interrupt. The entire Function Interrupt Status Bitmap is Read Only Zero if MSI Supported is Clear.

4.2.5. Misc. Device Configuration Space Requirements

4.2.5.1. BIST (Device)

BIST remains optional in MR-IOV. The results of invoking BIST in any non-BF Function must not affect any other VH.

The results are undefined if software invokes BIST in a BF when any VC to VL Map Enable bit in any Function Table entry of any BF of the Device is Set.

4.2.5.2. Device PCIe Capability Fields

Behavior of certain fields in the PCI Express Capability changes due to virtualization. Unless indicated in Table 4-26, behavior is as specified in the *PCI Express Base Specification*.

Table 4-26: Device PCIe Capability Fields

Register	Field(s)	VH0 Attributes	VHn Attributes
Device Capabilities	Captured Slot Power Limit Value Captured Slot Power Value	Contains the value from the most recent Set Slot Power message received. For an MR Link, Set Slot Power messages on any VH will update this field. Default value is 0.	
Device Capabilities	Function Level Reset Capability	Must be 1b.	
Device Control	Max_Payload_Size	Fully Implemented. PCIe Base rules apply within each VH. (See Implementation Note below).	
Device Control	Auxiliary (AUX) Power PM Enable	Component is allowed to draw AUX power if at least one of the Functions, in any VH, has this bit set.	
Device Status	Aux Power Detected	Implemented if Function supports AUX power.	
Link Capabilities	Supported Link Speeds Maximum Link Width ASPM Support	Reflect the physical link values.	
Link Control	ASPM Control Common Clock Configuration Extended Sync Enable Clock Power Management Hardware Autonomous Width Disable	Implemented	R/W fields with no effect
Link Status	Current Link Speed Negotiated Link Width Link Training Slot Clock Configuration	Reflect the physical link values.	
Device Capabilities 2	Completion Timeout Ranges Supported Completion Timeout Disable Supported	Implemented in each VH	
Device Control 2	Completion Timeout value Completion Timeout Disable	Implemented in each VH	

Register	Field(s)	VH0 Attributes	VHn Attributes
Link Control 2	Target Link Speed Enter Compliance Hardware Autonomous Bandwidth Disable	Implemented	R/W fields with no effect
Link Status 2	Current De-emphasis Level	Implemented	Returns the VH0 value

IMPLEMENTATION NOTE

Managing Maximum Payload Size

The Switch register Maximum Payload Size Offered (Section 4.3.6.2) can be used to restrict the Maximum Packet Size of a particular VH. In PCIe, system software must set Device values to maximum Payload Size to be compatible with the value set in attached Switches. By controlling the Maximum Payload Size Offered values in Swtiches, MR-PCIM can indirectly control the values used by Devices attached to those Switches.

4.3. Switch Configuration Space

For managing MR-IOV Switches, MR-PCIM must use the Switch Type 1 headers to manage the Switch. Configuration controls are associated with:

- Entire component
- Physical Port of the component
- Virtual Switch within the component
- PCI-to-PCI Bridges within each Virtual Switch

There are nine tables provided by the Switch. These tables are located using the MR-IOV Capability block located in the Type 1 Configuration Space of upstream P2P Bridge(s). An overview of the tables is shown in Figure 4-5.

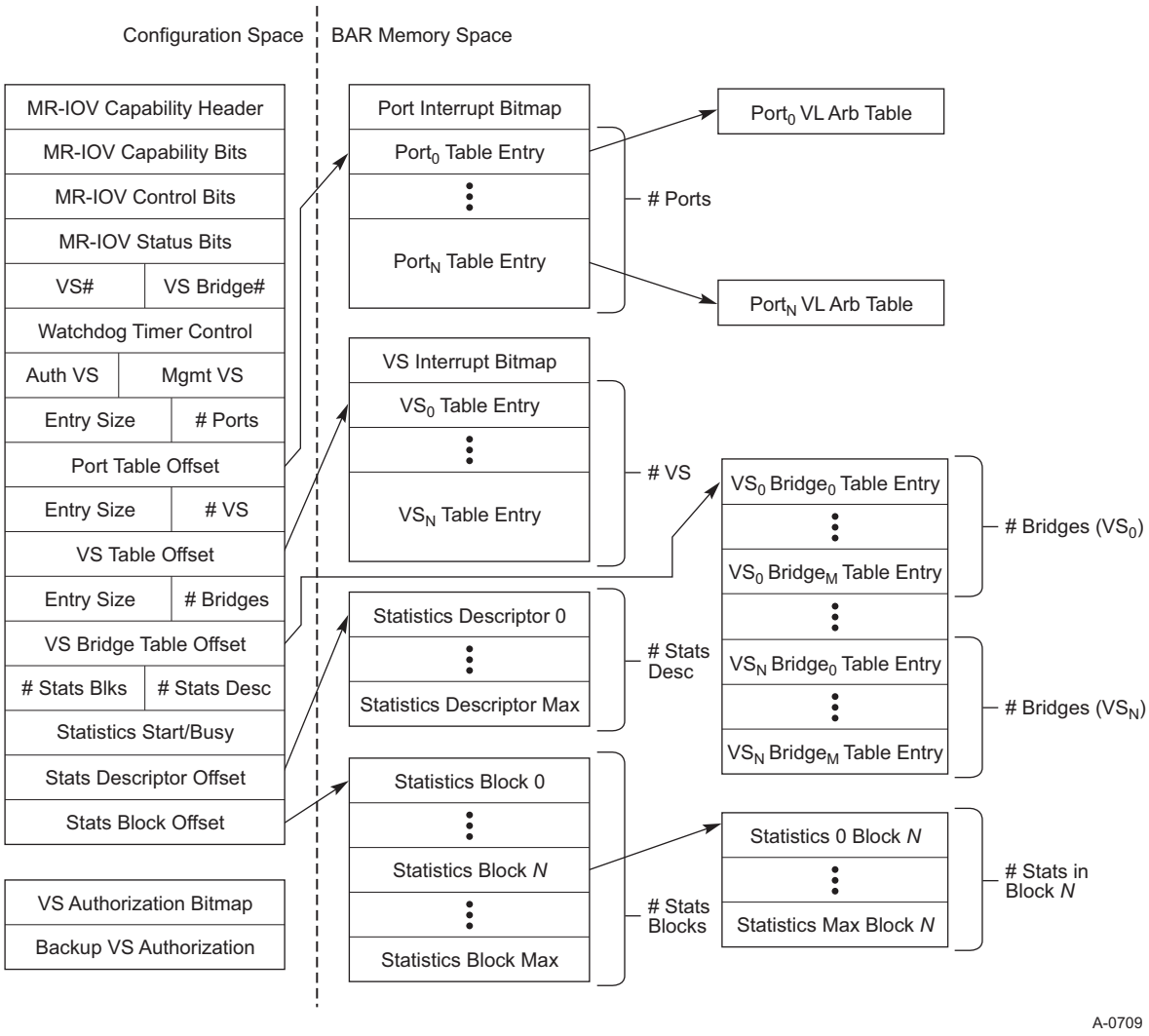

Figure 4-5: Switch Mapping Tables

- The MR-IOV Capability contains the controls that apply to the entire Switch. It also contains BAR relative offsets to tables located in BAR Memory Space and associated table size information.
- The VS Authorization Bitmap contains one bit for each Virtual Switch. If the corresponding bit is Set, the associated VS is authorized and can be used to mange the MR Switch.
- The Port Table contains an entry for every Port on the Switch. The table may be sparse (i.e., contain unused table entries) for hardware implementation flexibility. The Port Table contains the fields that control the physical Link. The Port table also points to the optional VL Arbitration Table. Preceding the Port Table is a 256-bit Port Interrupt Summary.
- The optional VL Arbitration Table supports controlling the arbitration between Virtual Links for access to a Port. This table is modeled after the VC Arbitration Table in PCI Express.

- The VS Table contains an entry for every Virtual Switch in the MR Switch. The table may be sparse for hardware implementation flexibility. Preceding the VS Table is a 256-bit VS Interrupt Summary.
- The VS Bridge Table contains an entry for every PCI-to-PCI Bridge in every Virtual Switch. This table is a two dimensional array indexed by VS number and by P2P Bridge number. Within a VS, the first entry corresponds to the upstream P2P Bridge and the remaining entries are the downstream bridge(s). This table may also be sparse, but the upstream entry of each VS must be present if the associated VS is present. No entries are present in this table unless the associated VS table is also present.
- The optional Statistics Descriptor Table contains descriptions of the varieties of performance counters and statistics information supported. This table is read only and contains one entry for each counter style supported by the component.
- The optional Statistics Block Table contains an entry for every block of related statistics counters. Each entry contains controls for the block and an offset to the array of counters.
- The optional Statistics Counter Table contains the actual counters and sampled values. There is one table for each Statistics Block. The offset and size of this table is contained in the associated Statistics Block Table.

These tables must be visible in the Upstream P2P Bridge of all authorized VHs. A subset of the MR-IOV Capability (but none of the other tables) may optionally be present in the Upstream P2P Bridges of non-authorized VHs. This subset MR-IOV Capability must be present in VH 0 Upstream P2P Bridges that are attached to MR aware Root Ports.

4.3.1. Switch MR-IOV Extended Capability

Figure 4-6 shows the Switch MR-IOV Extended Capability structure in more detail.

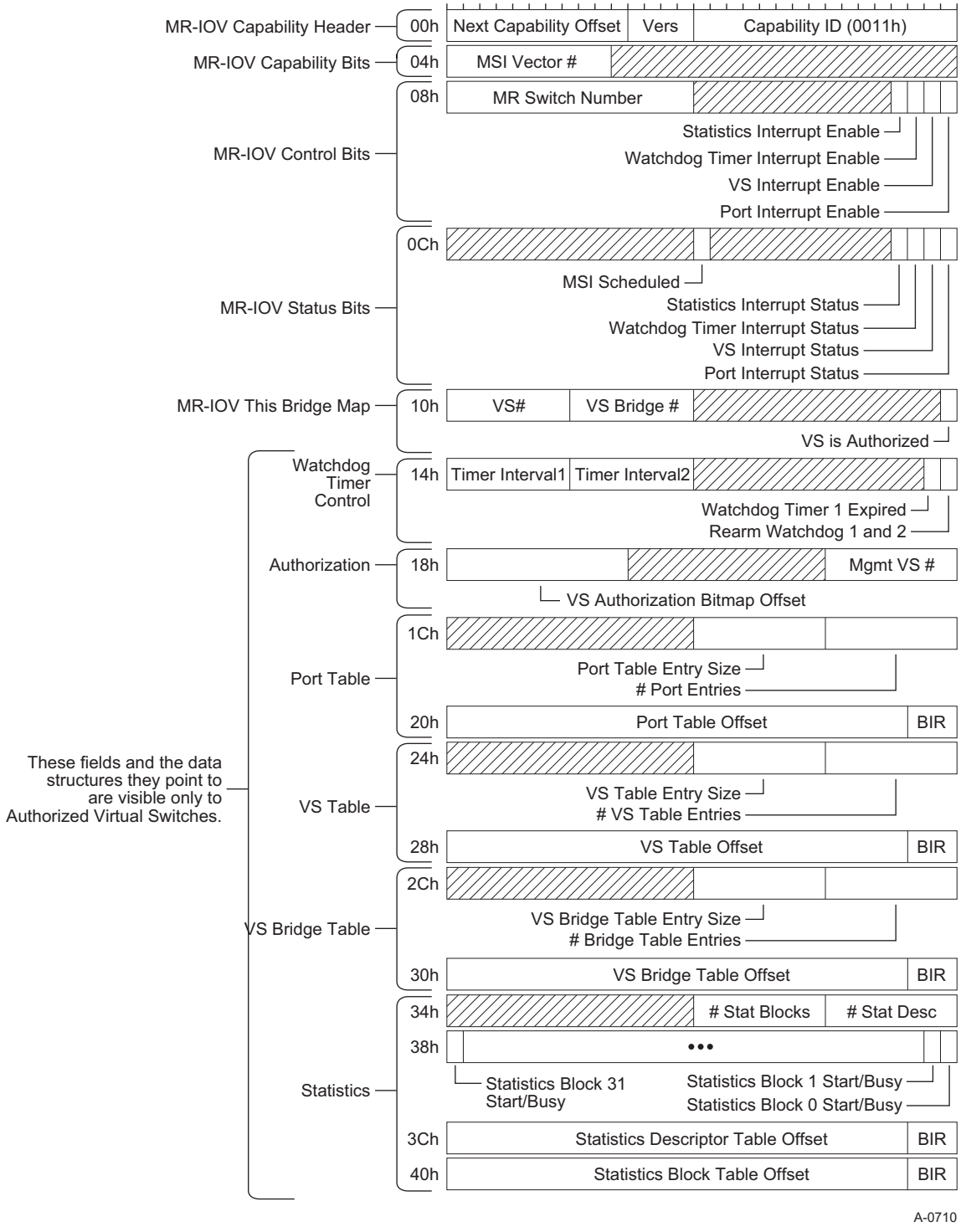

Figure 4-6: Switch MR-IOV Capability Diagram

4.3.1.1. Switch MR-IOV Extended Capability Header (00h)

Table 4-27 defines the Switch MR-IOV Extended Capability header. The Capability ID for the Switch MR-IOV Extended Capability is 0011h.

Table 4-27: Switch MR-IOV Extended Capability Header

Bit Location	Register Description	Attributes
15:0	**PCI Express Extended Capability ID** – This field is a PCI-SIG defined ID number that indicates the nature and format of the Extended Capability. The Extended Capability ID for the MR-IOV Extended Capability is 0011h.	RO
19:16	**Capability Version** – This field is a PCI-SIG defined version number that indicates the version of the Capability structure present. Must be 1h for this version of the specification.	RO
31:20	**Next Capability Offset** – This field contains the offset to the next PCI Express Capability structure or 000h if no other items exist in the linked list of Capabilities. This offset is relative to the beginning of PCI compatible Configuration Space and thus must always be either 000h (for terminating list of Capabilities) or greater than 0FFh.	RO

4.3.1.2. Switch MR-IOV Capability (04h)

Table 4-28: Switch MR-IOV Capability Bits

Bit Location	Register Description	Attributes
21:0	**Reserved**	RO
31:21	**MSI Vector Number** – This field indicates the MSI Vector number used to signal MR-IOV events within this Switch. This value may change based on whether MSI or MSI-X is enabled.	RO

4.3.1.3. Switch MR-IOV Control (08h)

Table 4-29: Switch MR-IOV Control Bits

Bit Location	Register Description	Attributes
0	**Port Interrupt Enable** – Enables delivery of Port Interrupts. See the Port Table for details. This bit is implemented per MR-IOV capability. This bit is Read Only Zero unless this VS is Authorized. The default value of this field is 0b.	RW
1	**VS Interrupt Enable** – Enables delivery of VS Interrupts. See the VS Table for details. This bit is implemented per MR-IOV capability. This bit is Read Only Zero unless this VS is Authorized. The default value of this field is 0b.	RW
2	**ReservedZ**.	RO
3	**Statistics Interrupt Enable** – Enables delivery of Statistics Interrupts. This bit is Read Only Zero unless this VS is Authorized or is Authorized in the Backup Authorization bitmap. This bit is not implemented and Read Only Zero if the Number of Statistics Blocks is 0 (see Section 4.5.1.1). The default value of this field is 0b.	RW
15:4	**ReservedP**	RO
31:16	**MR Switch Number** – Scratchpad register used by MR-PCIM in detecting loops during Topology Enumeration. This field is Read Only unless this VS is Authorized.	RW

4.3.1.4. Switch MR-IOV Status (0Ch)

Table 4-30: Switch MR-IOV Status Bits

Bit Location	Register Description	Attributes
0	**Port Interrupt Status** – Indicates that a Port Interrupt is pending. This bit is Read Only and indicates that some bit in the Port Interrupt Bitmask is Set. This bit is Read Only Zero if this VS is not Authorized.	RO
1	**VS Interrupt Status** – Indicates that a VS Interrupt is pending. This bit is Read Only and indicates that some bit in the VS Interrupt Bitmask is Set. This bit is Read Only Zero if this VS is not Authorized.	RO
2	**ReservedZ**	RO
3	**Statistics Interrupt Status** – This bit is Set when a Statistics Collection process completes. This bit is Read Only Zero if the Number of Statistics Blocks is zero (see Section 4.5.1.1)	RW1C
14:4	**ReservedZ**	RO
15	**MSI Scheduled** – Set when an MSI has been requested. If Set, subsequent MSIs are suppressed. If Clear, any enabled interrupt will cause an MSI to be scheduled (and this bit to be Set). The default value of this bit is 0b.	RW1C
31:16	**ReservedZ**	RO

4.3.1.5. MR-IOV This Bridge Map (10h)

This Read Only register returns the VS number and P2P Bridge number within the VS of this Type 1 Configuration header. In the upstream P2P Bridge of a VS, it also indicates the Authorization status of the VS.

Table 4-31: Switch MR-IOV This Bridge Map

Bit Location	Register Description	Attributes
0	**VS is Authorized** – Indicates that the VS is authorized to manage the Switch. See Authorization Control for details.	RO
15:1	**ReservedZ**	RO
23:16	**VS Bridge Number** – Indicates the P2P Bridge number within the indicated VS of this Type 1 Configuration Space. This value can be used to locate the associated VS Bridge Table Entry.	RO
31:24	**VS Number** – Indicates the VS number of this Type 1 Configuration Space. This value can be used to locate the associated VS Table and VS Bridge Table Entries.	RO

4.3.1.6. Watchdog Timer Control (14h)

The Watchdog Timer is used ensure that a backup MR-PCIM can recover and take over from the primary MR-PCIM. One key area is the situation where discovery has failed and the Initial MR-PCIM has altered the Switch configuration(s) such that the backup MR-PCIM cannot manage a Switch (e.g., the Backup MR-PCIM VS could have been de-authorized, some inter-Switch Link directions could be configured inappropriately, etc.).

Table 4-32: Switch MR-IOV Authorization Control

Bit Location	Register Description	Attributes
0	**Rearm Watchdog**– Writing 1 causes the Watchdog Timer to start counting. Writing 0 has no effect. Reads as zero. This field is Read Only Zero and writes have no effect if this VS is not Authorized.	RW
15:1	**ReservedZ**	RO
23:16	**Watchdog Timer Interval** – Determines the duration of Watchdog Timer. The timer expires if this period of time elapses before software restarts the watchdog timer. If the timer expires, the MR Switch is returned to its Initial Power-On Condition (e.g., parameters are reloaded based on straps, EEPROM, etc.). Duration of the timer is this field times 1 second (+50% -0%). The value 0 indicates the timer is disabled and will never expire. Default is 0h (i.e., disabled). This field is Read Only Zero if this VS is not Authorized.	RW
31:24	**ReservedZ**	RO

4.3.1.7. Authorization (18h)

This register determines which Virtual Switches are authorized to manage the MR Switch. It also indicates which VS receives "route to MR-PCIM" messages initiated by this Switch.

Table 4-33: Switch MR-IOV Authorization Control

Bit Location	Register Description	Attributes
7:0	**Management VS** – Indicates which VS is considered the primary management VS. The active MR-PCIM is running above the Root Port at the top of the hierarchy containing this VS. The indicated VS is automatically authorized independent of the state of the corresponding VS Authorization Bitmap entry. This field is Read Only Zero if this VS is not Authorized.	RW
19:8	**ReservedP**	RO
31:20	**VS Authorized Bitmap Offset** – This field contains the offset to the Authorization Bitmap. This offset is relative to the beginning of PCI compatible Configuration Space. See Section 4.3.2 for details.	RO

4.3.1.8. Port Table Entry Size/Num Port Entries (1Ch)

Table 4-34: Switch Port Table Sizes

Bit Location	Register Description	Attributes
7:0	**Num_Port_Table_Entries** – Returns the number of entries in the Port Table.	RO
15:8	**Port_Table_Entry_Size** – Returns the size of a Port Table Entry in DWORDs. For the current version of this specification, this value must be 15h or larger. Implementations may use larger values to simplify address arithmetic.	RO
31:16	**ReservedZ**	RO

The total size of the Port Table (in bytes) is:

Num_Port_Table_Entries * Port_Table_Entry_Size * 4

4.3.1.9. Port Table Offset (20h)

Table 4-35: Switch Port Table Offset

Bit Location	Register Description	Attributes
2:0	**Port Table BIR** – Indicates which one of a function's Base Address registers, located beginning at 10h in Configuration Space, is used to map the Function's Port Table into Memory Space. **BIR Value Base Address register** 0 BAR0 10h 1 BAR1 14h 2..7 Reserved For a 64-bit Base Address register, the BIR indicates the lower DWORD.	RO
31:3	**Port Table Offset** – Used as an offset from the address contained by one of the Function's Base Address registers to point to the base of the Port Table. The lower 3 BIR bits are masked off (set to zero) by software to form a 32-bit offset that is QWORD aligned. The minimum value of this field is 4 (the Port Interrupt Status Bitmap is 20h bytes and preceeds the Port Table, see Section 4.3.3.13).	RO

The Port Table starts at the Port Table Offset. The Port Interrupt Bitmap immediately precedes the Port Table (i.e., it starts 32 bytes before the Port_Table_Offset).

4.3.1.10. VS Table Entry Size/Num VS Table Entries (24h)

Table 4-36: Switch VS Table Sizes

Bit Location	Register Description	Attributes
7:0	**Num_VS_Table_Entries** – Returns the number of entries in the VS Table.	RO
15:8	**VS_Table_Entry_Size** – Returns the size of a VS Table Entry in DWORDs. For the current version of this specification, this value must be 3h or larger (more if some VS supports more than 32 Bridges). Implementations may use larger values to simplify address arithmetic.	RO
31:16	**Reserved**	RO

The total size of the VS Table (in bytes) is:

Num_VS_Table_Entries * VS_Table_Entry_Size * 4

4.3.1.11. VS Table Offset (28h)

Table 4-37: Switch VS Table Offset

Bit Location	Register Description	Attributes
2:0	**VS Table BIR** – Indicates which one of a function's Base Address registers, located beginning at 10h in Configuration Space, is used to map the Function's VS Table into Memory Space. **BIR Value Base Address register** 0 BAR0 10h 1 BAR1 14h 2..7 Reserved For a 64-bit Base Address register, the BIR indicates the lower DWORD.	RO
31:3	**VS Table Offset** – Used as an offset from the address contained by one of the Function's Base Address registers to point to the base of the VS Table. The lower 3 BIR bits are masked off (set to zero) by software to form a 32-bit offset that is QWORD aligned. The minimum value of this field is 4 (the VS Interrupt Status Bitmap is 20h bytes and preceeds the VS Table, see Section 4.3.5.4).	RO

The VS Table starts at the VS Table Offset. The VS Interrupt Bitmap immediately precedes the VS Table (i.e., it starts 32 bytes before the VS_Table_Offset).

4.3.1.12. VS Bridge Table Entry Size/Num VS Bridge Table Entries per VS (2Ch)

Table 4-38: Switch VS Bridge Table Sizes

Bit Location	Register Description	Attributes
7:0	**Num_Bridge_Table_Entries** – Returns the number of Bridge entries in the VS Bridge Table associated with each VS.	RO
15:8	**Bridge_Table_Entry_Size** – Returns the size of a VS Bridge Table Entry in DWORDs. For the current version of this specification, this value must be at least 8. Implementations may use larger values to simplify address arithmetic.	RO
31:16	**Reserved**	RO

The total size (in bytes) of the VS Bridge Table is:

Num_Bridge_Table_Entries * Num_VS_Table_Entries * Bridge_Table_Entry_Size * 4

Note: Num_Bridge_Table_Entries reflects the size of the table and thus is the maximum size across all Virtual Switches. Not all VS Bridge Table Entries need be present (see Section 4.3.6.1).

4.3.1.13. VS Bridge Table Offset (30h)

Table 4-39: Switch VS Bridge Table Offset

Bit Location	Register Description	Attributes
2:0	**VS Bridge Table BIR** – Indicates which one of a function's Base Address registers, located beginning at 10h in Configuration Space, is used to map the Function's VS Bridge Table into Memory Space. **BIR Value Base Address register** 0 BAR0 10h 1 BAR1 14h 2..7 Reserved For a 64-bit Base Address register, the BIR indicates the lower DWORD.	RO
31:3	**VS Bridge Table Offset** – Used as an offset from the address contained by one of the Function's Base Address registers to point to the base of the VS Bridge Table. The lower 3 BIR bits are masked off (set to zero) by software to form a 32-bit offset that is QWORD aligned.	RO

The VS Bridge Table Entry associated with Bridge 0 of VS *N* immediately follows the last Bridge Table Entry associated with VS *N-1*.

4.3.1.14. Statistics Capability and Control (30h to 3Ch)

Device and Switch Statistics related fields are described in Section 4.5.

4.3.2. Switch VS Authorization Bitmap

The VS Authorization Bitmap is located in Configuration Space at the byte offset indicated by the VS Authorization Bitmap Offset. This table must be located in Extended Configuration Space (i.e., the offset must be greater than 0FFh).

The table contains Num_VS_Table_Entries bits. Bit 0 of the first DWORD corresponds to VS 0, bit 1 corresponds to VS 1, etc. A VS is authorized if either the corresponding bit in the VS Authorization Bitmap is Set or if the VS is the Management VS. The Bitmap contains an integral number of DWORDS. Unused bits in the last DWORD and bits corresponding to VSs where VS Present is 0b are read only zero.

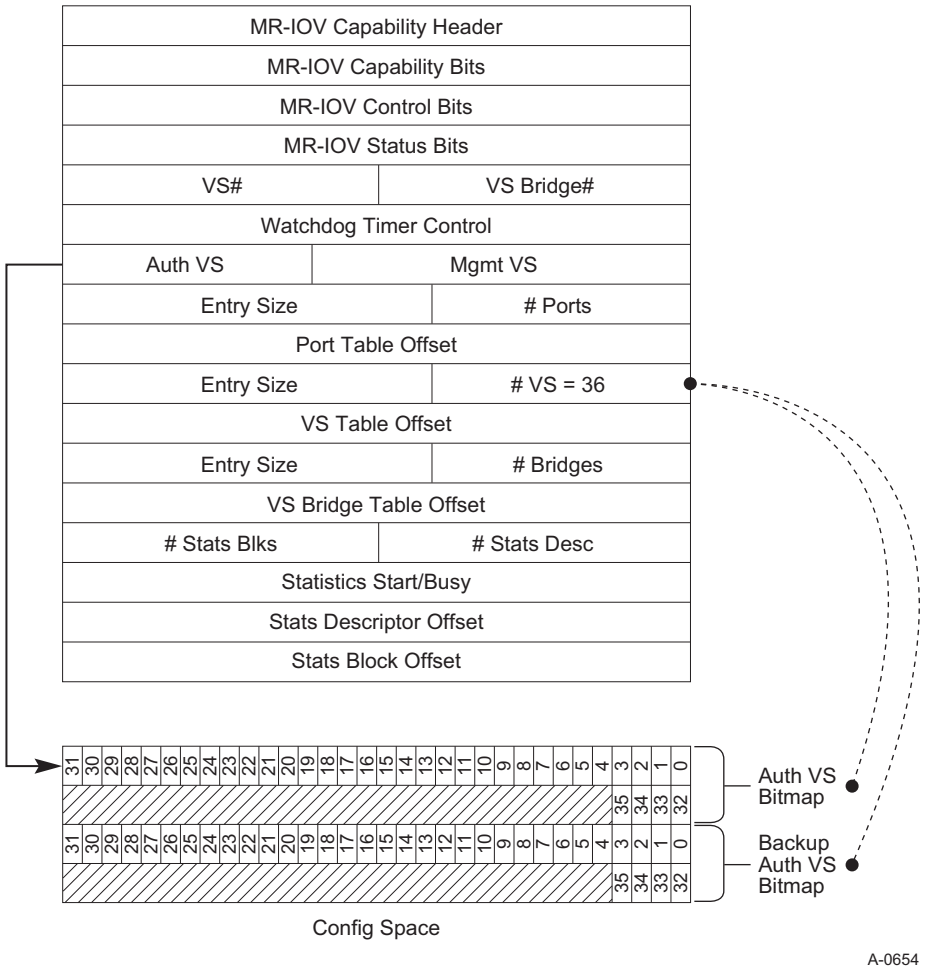

Figure 4-7: Example Authorization Bitmap (# VS = 36)

Only transactions from Authorized VSs are allowed to access Memory Space MR-IOV tables of this Switch.

Only transactions from Authorized VSs are allowed to access certain fields Configuration Space of the MR-IOV Extended Capability.

The Authorization Bitmap is Read Only Zero if this VS is not Authorized.

Note: The bitmap entry for the Management VS is read/write but its value does not affect Authorization state of that VS. If the "old" Management VS is to remain Authorized after a transition to a "new" Management VS, software should Set the bitmap entry for the "old" VS before changing Management VS to ensure that Authorization remains seamless.

4.3.3. Switch Port Table

The Port Table contains up to 256 Port Table Entries. This table is located in memory starting at a location determined by the Port Table Offset field in the MR-IOV Capability. This table is Read Only Zero unless the associated VS is Authorized.

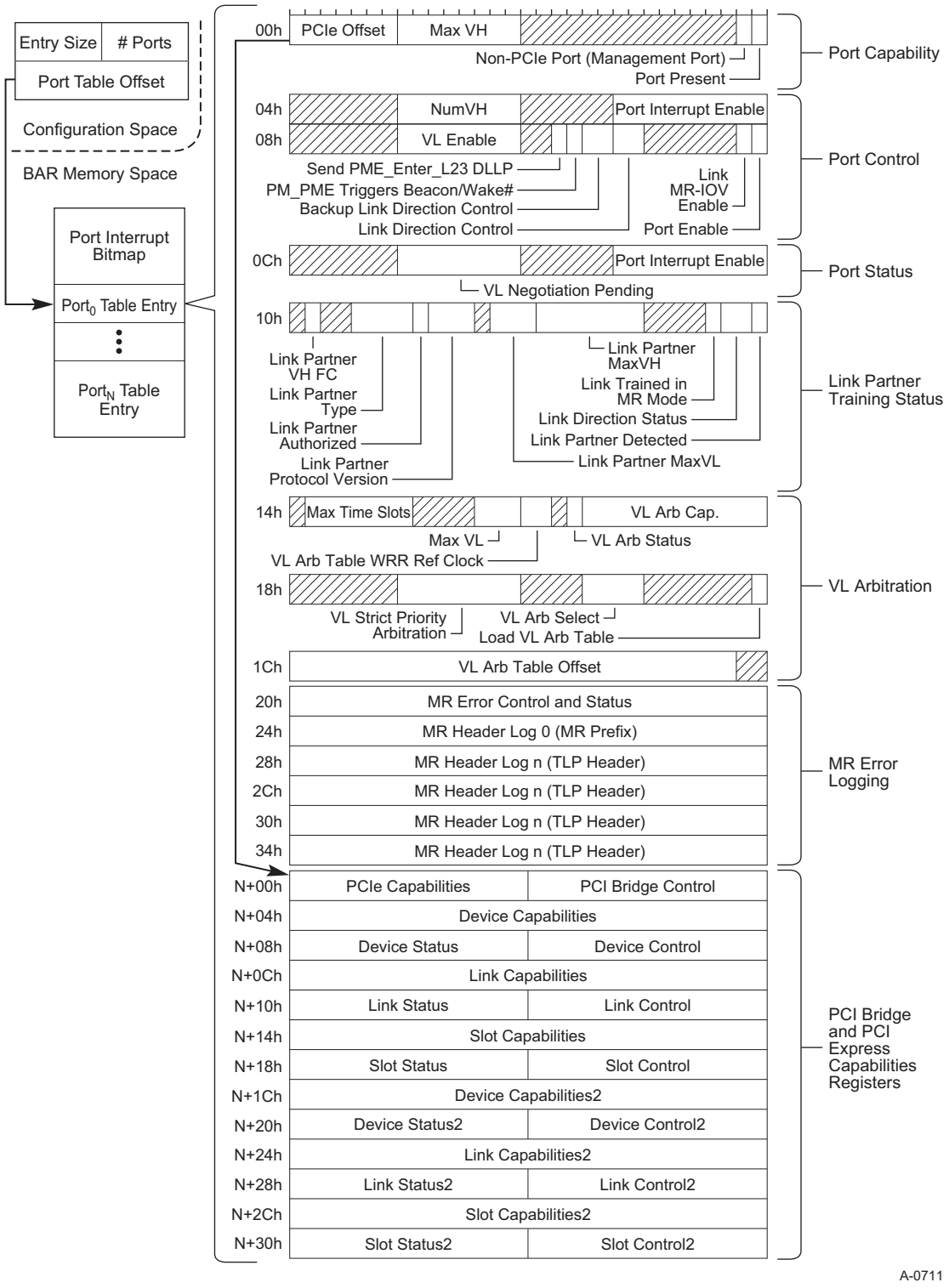

Figure 4-8: Switch Port Table

4.3.3.1. Port Capability (00h)

Table 4-40: Switch Port Capability

Bit Location	Register Description	Attributes
0	**Port Present** – Indicates that this Port is present on the Switch. If Clear, the contents of this Port Table Entry are undefined. This bit allows the Port Table to be sparse.	RO
1	**Non-PCIe Switch Management Port** – Indicates that this is a Vendor Specific non-PCI Express Port used to manage a Switch (see Section 3.1.1.3). Non-PCIe Switch Management Ports are always Upstream Ports that are not MR Capable. Unless otherwise specified, fields in the Port Table Entry are Read Only Zero. The Port Interrupt Status bit associated with Port is always Clear and the Port will not generate a Port Interrupt[13].	RO
2	**Link MR Capable** – Indicates that a Port is capable of supporting MR Operation. If Set, the Link MR-IOV Enable field is implemented. Must be 0b if Non-PCIe Switch management Port is Set.	RO
5:3	**Link Direction Supported** – Indicates what values of Link Direction Control are supported. Bit 0 Link Direction Control can be Upstream Component Bit 1 Link Direction Control can be Downstream Component Bit 2 Link Direction Control can be Crosslink Must be 001b if Non-PCIe Switch management Port is Set.	RO
7:6	**Port Direction Supported** – Indicates what values of Port Direction Control are supported. Values are: 00b Reserved 01b Port Direction Control can be Upstream Component 10b Port Direction Control can be Downstream Component 11b Port Direction Control can be either an Upstream or a Downstream Component Must be 10b if Non-PCIe Switch Management Port is Set.	RO
15:8	**Reserved**	RO
23:16	**MaxVH** – Maximum number of VHs supported on this Port minus 1. Must be Zero if Link MR Capable is Clear.	RO
31:24	**PCIe Offset** – DWORD Offset to the PCIe Capabilities section of the Port Table Entry. For the current version of the specification, this must be at least 0Eh (i.e., 38h divided by 4).	RO

[13] If interrupts are needed, additional MSI Vectors can be used.

4.3.3.2. Port Control (04h and 08h)

Table 4-41: Switch Port Control1

Bit Location	Register Description	Attributes
0	**Port DL_Up Interrupt Enable** – When both this bit and the Port DL_Up Interrupt Pending bit are set, the Port Interrupt Status bit associated with this Port is Set and a Port Interrupt is requested. Default is 0b.	RW
1	**Port DL_Down Interrupt Enable** – When both this bit and the Port DL_Down Interrupt Pending bit are set, the Port Interrupt Status bit associated with this Port is Set and a Port Interrupt is requested. Default is 0b.	RW
2	**Port PCIe Capability Interrupt Enable** – When both this bit and the Port PCIe Capability Interrupt Pending bit are set, the Port Interrupt Status bit associated with this Port is Set and a Port Interrupt is requested. Default is 0b.	RW
3	**Link Retrain Interrupt Enable** – When both this bit and the Link Retrain Interrupt Pending bit are set, the Port Interrupt Status bit associated with this Port is Set and a Port Interrupt is requested. Default is 0b.	RW
4	**Beacon/WAKE# Interrupt Enable** – When both this bit and the Beacon/WAKE# Interrupt Pending bit are set, the Port Interrupt Status bit associated with this Port is Set and a Port Interrupt is requested. Default is 0b. Beacon/WAKE# support is optional. If not supported, this bit is Read Only Zero.	RW
5	**MR Uncorrectable Fatal TLP Error Interrupt Enable** – When both this bit and the MR Uncorrectable Fatal Error Status bit are Set, the Port Interrupt Status bit associated with this Port is Set and a Port Interrupt is requested. Default is 0b.	RW
6	**MR Uncorrectable Non-Fatal TLP Error Interrupt Enable** – When both this bit and the MR Uncorrectable Non-Fatal Error Status bit are Set, the Port Interrupt Status bit associated with this Port is Set and a Port Interrupt is requested. Default is 0b.	RW
7	**MR Uncorrectable Global Key Error Interrupt Enable** – When both this bit and the MR Uncorrectable Global Key Error Status bit are set, the Port Interrupt Status bit associated with this Port is Set and a Port Interrupt is requested. Default is 0b.	RW
8	**MR Correctable Global Key Error Interrupt Enable** – When both this bit and the MR Correctable Global Key Error Status bit are Set, the Port Interrupt Status bit associated with this Port is Set and a Port Interrupt is requested. Default is 0b. This field is Read Only Zero if VS Global Key Entering Check Supported and VS Global Key Exiting Check Supported are both Clear.	RW

Bit Location	Register Description	Attributes
9	**MR DLLP Error Interrupt Enable** – When both this bit and the MR DLLP Error Status bit are Set, the Port Interrupt Status bit associated with this Port is Set and a Port Interrupt is requested. Default is 0b.	RW
10	**Physical Hot-Plug Interrupt Enable** – When both this bit and the Physical Hot-Plug Interrupt Pending bit are set, the Port Interrupt Status bit associated with this Port is Set and a Port Interrupt is requested. Default is 0b.	RW
15:11	**Reserved**	RsvdP
23:16	**NumVH** – Indicates the number of VHs enabled. This value must be less than or equal to MaxVH. The default value of this field is Vendor Specific. This value may be used by the Switch to optimize resource usage. This field is Read Only if MR Enable is Set. This field is Read/Write if MR Enable is Clear.	RW
30:24	**Reserved**	RO
31	**MR Enable** – If Set, NumVH is Read Only and software may enable additional VLs and VHs. If Clear, NumVH may be written and additional VHs and VLs cannot be enabled. Default is Vendor Specific. Hardware behavior on the 1b to 0b transition of this field when LinkUp is 1b is undefined.	RW

Table 4-42: Switch Port Control2

Bit Location	Register Description	Attributes
0	**Port Enable** – Set by MR-PCIM to indicate that it wishes to use the Port.	RW
1	**Link MR-IOV Enable** – If Set, the Link will attempt to negotiate to use the MR-IOV enhanced protocol. If Clear, the Link will not attempt to use MR-IOV and will thus train the Link in Base PCIe mode. Read Only if the Link is up as indicated by Link Direction Status. Hardwired to 0b if Link MR Capable is Clear. Default is Vendor Specific. This value may be changed at any time, but the change takes effect only when the link is in Detect.Quiet. The default value for most MR systems should be 1b. The default for an MR Switch configured to operate as a Base PCIe Switch should be 0b. Note: PCIe requires that components ignore unsupported DLLPs. As such, PCIe components should train in Base PCIe mode independent of the value of this bit. For components where this is not true, this bit can be cleared as a workaround.	RW
7:2	**Reserved**	RO
9:8	**Link Direction Control** – Controls how the Link should train when leaving LTSSM state Detect.Quiet. Values are: 0 Train as Upstream Component 1 Train as Downstream Component 2 Train as Crosslink 3 Do not train, remain in Detect.Quiet and keep Link down Writing Link Direction Control to a value that is incompatible with the value of Link Direction Supported is undefined. The default value of this field is Device Specific.	RW
11:10	**Reserved**	RO
12	**PM_PME Triggers Beacon/WAKE#** – If Set, the automatic triggering of Beacon/WAKE# on reception of a Beacon, WAKE# or PM_PME message at this Port is suppressed. See Section 7.6 for details.	RW
13	**Send PME_Enter_L23 DLLP** – If software writes 1b to this bit and Port Direction Control is 0b (indicating Upstream Switch Port), initiate the PME_Enter_L23 handshake to power down the Link. Writing 0 to this bit has no effect. This bit always returns 0b when read. If Port Direction Control is 1b, writing this bit has no effect. If Bridge Controls Physical Link is Set in an Upstream VS Bridge that is mapped to this Port, PME_Enter_L23 DLLPs are sent using PCIe rules applied within the associates VS and this bit has no effect.	Write 1 to Send

Bit Location	Register Description	Attributes
14	**Port Direction Control** – Controls whether, for PHY purposes, this port represents an Upstream Component or a Downstream Component. The value 0b represents an Upstream Component. The value 1b represents a Downstream Component. If Port Direction Supported is 01b, this bit is hardwired to 0b. If Port Direction Supported is 10b, this bit is hardwired to 1b. If Port Direction Supported is 11b, this bit is read/write and is updated to match the LTSSM Link Direction on entry to LTSSM state Configuration.Idle when LinkUp is 0b.	RW/RO
15	**Reserved**	RO
23:16	**VL Enable** – This bit, when Set, enables a Virtual Link (see Note 1 for exceptions). The Virtual Link is disabled when this bit is cleared. Software must use the VL Negotiation Pending bit to check whether the VL negotiation is complete. Default value of this bit is 1b for the first VL and is 0b for other VLs. Notes: 1. This bit is hardwired to 1b for the VL0; i.e., writing to this bit has no effect for VL0. 2. To enable a Virtual Link, the VL Enable bits for that Virtual Link must be Set in both components on a Link. 3. To disable a Virtual Link, the VL Enable bits for that Virtual Link must be cleared in both components on a Link. 4. Software must ensure that no traffic is using a Virtual Link at the time it is disabled. 5. Software must fully disable a Virtual Link in both components on a Link before re-enabling the Virtual Link.	RW
31:24	**Reserved**	RO

4.3.3.3. Port Status (0Ch)

Table 4-43: Switch Port Status

Bit Location	Register Description	Attributes
0	**Port DL_Up Interrupt Pending** – Set the Link enters DL_Up. This bit is Cleared when software writes 1b.	RW1C
1	**Port DL_Down Interrupt Pending** – Set the Link enters DL_Down. This bit is Cleared when software writes 1b.	RW1C
2	**Port PCIe Capability Interrupt Pending** – Set when an interrupt is sent due to status bits in the Port PCIe Capability Structure.	RW1C
3	**Link Retrain Interrupt Pending** – Set when the Link retrains. Can be used to monitor Link health.	RW1C
4	**Beacon/WAKE# Interrupt Pending** – Set when the Link detects a Beacon or WAKE# event. Beacon/WAKE# support is optional. If not supported, this bit is Read Only Zero. When supported, it is form factor specific whether Beacon or WAKE# is used.	RW1C
5	**MR Uncorrectable Fatal Error Interrupt Pending** – Set when the Link detects an MR Uncorrectable fatal Error and Sets one of the MR Error Status bits.	RW1C
6	**MR Uncorrectable Non-Fatal Error Interrupt Pending** – Set when the Link detects an MR Uncorrectable Non-Fatal Error and Sets one of the MR Error Status bits.	RW1C
7	**MR Correctable Error Interrupt Pending** – Set when the Link detects an MR Correctable Error and Sets one of the MR Error Status bits.	RW1C
8	**Physical Hot-Plug Interrupt Pending** – Set when the Physical Hot-Plug controller indicates that software should be notified.	RW1C
15:9	**Reserved**	RsvdZ
23:16	**VL Negotiation Pending** – These bits indicate whether Virtual Link Negotiation for some VL is in pending state. The value of these bits is defined only when the Link is in the DL_Active state and the Virtual Link is enabled (its VL Enable bit is Set). When these bits are Set by hardware, it indicates that the VL resource has not completed the process of negotiation. These bits are cleared by hardware after the VL negotiation is complete (on exit from the MR FC_INIT2 state on the VL). Before using a Virtual Link, software must check whether the VL Negotiation Pending bits for that Virtual Link are Clear in both components on the Link.	RO
31:24	**Reserved**	RsvdZ

4.3.3.4. Link Partner Training Status (10h)

Table 4-44: Switch Link Partner Training Status

Bit Location	Register Description	Attributes
0	**Link Partner Detected** – Set to indicate that a PCI Express component was detected at the remote end of the one or more Lanes of the Link.	RO
2:1	**Link Direction Status** – Indicates whether (and how) the Link trained. Values are: 0 LinkUp is 1b and Upstream Component 1 LinkUp is 1b and Downstream Component 2 LinkUp is 0b, LTSSM is not in Detect.Quiet 3 LinkUp is 0b, LTSSM is in Detect.Quiet This field is set to 0 on entry to LTSSM state Configuration.Idle when the LTSSM link direction indicates Upstream Component (Downstream Lanes). This field is set to 1 on entry to LTSSM state Configuration.Idle when the LTSSM link direction indicates Downstream Component (Upstream Lanes). This field is set to 2 whenever LinkUp is 0b and the LTSSM is in any state other than Detect.Quiet. This field is set to 3 whenever LinkUp is 0b and the LTSSM is in state Detect.Quiet. In all other LTSSM transitions, this field is unchanged. Note that Link Direction Status and Link Direction Control may disagree with each other if (1) Link Direction Control indicates crosslink or (2) Link Direction Control has changed but the Link has not transitioned through Detect.Quiet where the change can take effect. Software that wishes to reverse the link direction should (1) disable the link by setting Link Direction Control to 3, (2) cause the link to retrain by setting Link Retrain, (3) wait for the link to disable by waiting for Link Direction Status to indicate 3, (4) enable the link in the appropriate direction by setting Link Direction Control to 0, 1, or 2, and (5) wait for the Link to train by waiting for Link Direction Status to indicate 0 or 1.	RO
3	**Link Partner is MR** – Link was successfully brought up in MR-IOV mode.	RO
7:4	**Reserved**	RO
31:8	**MR Init DLLP Bits** – These bits were captured during Link training from bytes 1 to 3 of the MR Init DLLP that was sent by the Link Partner. These bits are not meaningful unless Link Partner is MR is Set. These fields allow MR-PCIM to know some information about the Link Partner without needing to read Configuration Space (which is not possible if the Link Partner is an MR Root Port since Configuration requests cannot flow Upstream). The bits are further described in Table 4-45.	RO

Table 4-45: Switch Link Partner Training Status – MRInit DLLP Bits

Bit Location	Register Description	Attributes
15:8	**Link Partner MaxVH** – Maximum number of VHs that the Link Partner can support.	RO
18:16	**Link Partner MaxVL** – Maximum number of VLs that the Link Partner can support.	RO
19	**Reserved**	RO
22:20	**Link Partner Protocol Version** – MR-IOV Protocol version supported by the Link Partner. For this version of the specification, this is the value 1h.	RO
23	**Link Partner was Authorized** – If the Link Partner is an MR Switch, this bit indicates that at the time of Link Training the VS associated with VH0 of this Link was allowed to manage the Switch (authorization can be revoked so this may no longer be accurate). This corresponds to the VS is Authorized bit in the Link Partner's MR-IOV Capability.	RO
27:24	**Link Partner Device/Port Type** – Indicates what kind of PCI Express Device is present in the Link Partner. Encoding is based on the Device/Port Type field in the PCI Express Capabilities (Offset 02h, Bits 7:4). See Section 2.1 for details.	RO
28	**Link Partner Mixed Device/Port Type** – Indicates that the Link Partner contains multiple Functions that have a variety of Device Types. See Section 2.1 for details.	RO
29	**Reserved**	RO
30	**Link Partner VH FC** – If Set, indicates the Link Partner supports per-VH and Per-VL Flow Control. If Clear, indicates that the Link Partner supports only per-VL Flow Control.	RO
31	**Reserved**	RO

4.3.3.5. VL Arbitration Capability and Status (14h)

Table 4-46: Switch VL Arbitration Capability and Status

Bit Location	Register Description	Attributes
11:0	**VL Arbitration Capability** – Indicates the types of VL Arbitration supported by the Port. This field is valid for all Functions that report a Low Priority Extended VC Count field greater than 0. For all other Functions, this field must be hardwired to 00h. Each bit location within this field corresponds to a VC Arbitration Capability defined below. When more than 1 bit in this field is Set, it indicates that the Port can be configured to provide different VC arbitration services. Defined bit positions are: Bit 0 Hardware fixed arbitration scheme, e.g., Round Robin Bit 1 Weighted Round Robin (WRR) arbitration with 32 phases Bit 2 WRR arbitration with 64 phases Bit 3 WRR arbitration with 128 phases Bit 4 Time-based WRR with 128 phases Bit 5 WRR Arbitration with 256 phases Bits 6-10 Reserved Bit 10 Vendor Defined VL Arbitration Scheme	RO
12	**VL Arb Status** – This bit indicates the coherency status of the VL Arbitration Table. This bit is valid only when the VL Arbitration Table is used. This bit is Set by hardware when any entry of the VL Arbitration Table is written to by software. This bit is cleared by hardware when hardware finishes loading values stored in the VL Arbitration Table after software sets the Load VL Arbitration Table bit. Default value of this bit is 0b.	RO
13	**Reserved**	RO
15:14	**Reference Clock** – Indicates the reference clock for Virtual Links that support time-based WRR VL Arbitration. This field is valid only if time-based WRR is supported. Defined encodings are: 00b 100 ns reference clock 01b – 11b Reserved	RO
18:16	**MaxVL** – Indicates the number of VLs supported minus 1. The Port supports VL_0 through VL_{MaxVL} inclusive.	RO
23:19	**Reserved**	RO

Bit Location	Register Description	Attributes
30:24	**Maximum Time Slots** – Indicates the maximum number of time slots (minus one) that are supported when configured for time-based WRR VL Arbitration. For example, a value of 000 0000b in this field indicates the supported maximum number of time slots is 1 and a value of 111 1111b indicates the supported maximum number of time slots is 128. This field is valid only when the VL Arbitration Capability field indicates that time-based WRR VL Arbitration is supported.	RO
31	**Reserved**	RO

4.3.3.6. VL Arbitration Control (18h)

Table 4-47: Switch VL Arbitration Control

Bit Location	Register Description	Attributes
0	**Load VL Arbitration Table** – When Set, this bit updates the VL Arbitration logic from the VL Arbitration Table. This bit is valid only when the VL Arbitration Table is used by the selected VL Arbitration scheme (that is indicated by a Set bit in the VL Arbitration Capability field selected by VL Arbitration Select). Software sets this bit to signal hardware to update VL Arbitration logic with new values stored in VL Arbitration Table; clearing this bit has no effect. Software uses the VL Arbitration Table Status bit to confirm whether the new values of VL Arbitration Table are completely latched by the arbitration logic. This bit always returns 0b when read. Default value of this bit is 0b.	RW
7:4	**Reserved**	RO
11:8	**VL Arbitration Select** – This field configures the Port to provide a particular VL Arbitration service. The permissible value of this field is a number corresponding to one of the asserted bits in the VL Arbitration Capability field.	RO
15:12	**Reserved**	RO

Bit Location	Register Description	Attributes
23:16	**VL Strict Priority Arbitration** – This field contains one bit per VL. Bit 0 corresponds to VL0. Bit 7 corresponds to VL7. When a bit is Set, the corresponding VL is configured to arbitrate as Strict Priority based on VL number. When a bit is Clear, the corresponding VL is configured to arbitrate as normal priority (using the scheme selected by VL Arbitration Select). Among the VLs configured for strict priority, priority is based on increasing VL number. VL0 is the lowest strict priority; VL7 is the highest. Strict Priority VLs have priority over normal priority VLs. Behavior is Undefined if a VL configured for Strict Priority is also included in the VL Arbitration Table. If a VL is Disabled, the value of the corresponding bit in this field is ignored. Default value of this field is 0000 0000b.	RW
31:24	**Reserved**	RO

4.3.3.7. VL Arbitration Table Offset (1Ch)

Table 4-48: Switch VL Arbitration Table Offset

Bit Location	Register Description	Attributes
1:0	**Reserved**	RO
31:2	**VL Arbitration Table Offset** – DWORD Offset to the VL Arbitration Table	RO

4.3.3.8. MR Error Status (20h)

Table 4-49: Switch MR Error Status

Bit Location	Register Description	Attributes
3:0	**MR First Error Pointer** – The First Error Pointer is a field that identifies the bit position of the first error reported in the MR Error Status register.	RO
4	**MR Uncorrectable Fatal TLP Error Status** – This bit is Set when the Uncorrectable Fatal TLP Error Status is Set and the Uncorrectable Fatal TLP Error Mask bit is Clear.	RW1C
5	**MR Uncorrectable Non-Fatal TLP Error Status** – This bit is Set when the Uncorrectable Non-Fatal TLP Error Status is Set and the Uncorrectable Non-Fatal TLP Error Mask bit is Clear.	RW1C
6	**MR Uncorrectable Global Key Error Status** – This bit is Set when the Uncorrectable Global Key Error Status is Set and the Uncorrectable Global Key Error Mask bit is Clear.	RW1C
7	**MR Correctable Global Key Error Status** – This bit is Set when the Correctable Global Key Error Status is Set and the Correctable Global Key Error Mask bit is Clear. If the VS Global Key Entering Check Supported and the VS Global Key Exiting Check Supported fields are both Clear, this field is RsvdZ.	RW1C
8	**MR DLLP Error Status** – This bit is Set when the Uncorrectable Fatal TLP Error Status is Set and the Uncorrectable Fatal TLP Error Mask bit is Clear. Headers are not logged and the First Error Pointer is not updated for DLLP Errors.	RW1C
14:9	**Reserved**	RsvdZ
15	**MR Multiple Uncorrectable Error** – Set when hardware detects an MR Uncorrectable Error but is unable to indicate it because the Error Status bit is already Set.	RW1C

4.3.3.9. MR Error Control (22h)

Table 4-50: Switch MR Error Control

Bit Location	Register Description	Attributes
3:0	**Reserved**	RO
4	**MR Uncorrectable Fatal TLP Error Mask** – If this bit is Set, Uncorrectable Fatal TLP Errors are not logged and the MR Uncorrectable Fatal TLP Error Status will never be Set. The default is 0b.	RW
5	**MR Uncorrectable Non-Fatal TLP Error Mask** – If this bit is Set, Uncorrectable Non-Fatal TLP Errors are not logged and the MR Uncorrectable Non-Fatal TLP Error Status will never be Set. The default is 0b.	RW
6	**MR Uncorrectable Global Key Error Mask** – If this bit is Set, Uncorrectable Global Key Errors are not logged and the MR Uncorrectable Global Key Error Status will never be Set. The default is 0b.	RW
7	**MR Correctable Global Key Error Mask** – If this bit is Set, Correctable Global Key Errors are not logged and the MR Correctable Global Key Error Status will never be Set. The default is 0b. If the VS Global Key Entering Check Supported and VS Global Key Exiting Check Supported fields are both Clear, this field is RsvdP.	RW
8	**MR DLLP Error Mask** – If this bit is Set, Uncorrectable Fatal TLP Errors are not logged and the MR Uncorrectable Fatal TLP Error Status will never be Set. The default is 0b.	RW
15:9	**Reserved**	RsvdP

4.3.3.10. MR Header Log (24h to 34h)

These fields contain the TLP Prefix and TLP header corresponding to the error described by the First Error Pointer in the MR Error Status register.

The value of these fields is undefined if the First Error Pointer is zero or points to a bit number that is not Set.

Headers are not logged and the First Error Pointer is not updated for DLLP Errors.

A Switch may share Header Log Registers among Ports. A shared header log must have storage for at least one header.

When an MR error is detected in a Port, the error shall be logged. If a shared set of MR Header Log Registers is implemented, a Port may not have room to log a header. In this case, the Port shall update its MR Error Status Register, however, when that Port's MR Header Log Register is read, it shall return all 1's to indicate an overflow condition.

The Port's MR Header Log entry shall be locked and remain valid while that Port's First Error Pointer is valid. While the MR Header Log entry is locked, additional errors shall not overwrite the

locked entry for this or any other Port. When a MR Header Log entry is unlocked, it shall be available to record a new error for any Port sharing the header logs.

4.3.3.11. PCI Bridge Control (N+00h)

These fields are the PCI Bridge controls that affect the Physical Port.

Table 4-51: PCI Bridge Control

Bit Location	Register Description	Attributes
0	**Secondary Bus Reset** – If the Link is a Downstream PCIe Link setting, this bit causes the Port to initiate a PCI Express Hot Reset (TS1 with the Hot Reset bit Set). Clearing this bit causes the Port to remove the Hot Reset and attempt to bring the Link back up. Flow Control is renegotiated after TS1 style Hot Reset is removed. If Bridge Control Physical Link is Set, this bit is identical to the Secondary Bus Reset bit in the PCI Bridge Control register. If the Link Direction Status does not equal 1b (i.e., Link is Upstream or Down) or if the Link Partner Is MR bit is 1b, this bit must be 0b.	RW
15:0	**Reserved**	RsvdP

4.3.3.12. PCIe Capability Structure (N+02h)

These fields are the PCI Express controls that affect the Physical Port. Values in various Configuration Spaces are either Virtual values or map to these values (see the Bridge Controls Physical bit described in Section 4.3.6.2).

The layout of this structure is similar to the PCI Express Capability dropping the Root Capability, Control, and Status words. All fields are implemented as defined in the *PCI Express Base Specification* except as indicated in Table 4-52.

The Port PCIe Capability Interrupt Pending bit is set whenever the *PCI Express Base Specification* signals an interrupt based on status bits in this structure.

If the *PCI Express Base Specification* indicates a field has different behavior for Upstream vs Downstream Ports, the Port Direction Control field is used to determine the behavior of the field.

Table 4-52: Port PCIe Capability Structure

Register	Field(s)	Attributes
PCI Express Capabilities	Slot Implemented	HwInit. Reflects Physical Hot-Plug.
PCI Express Capabilities	Interrupt Message Number Device/Port Type	Not Implemented, Read Only Zero
Device Capabilities	Phantom Functions Supported	Must be 00b

Register	Field(s)	Attributes
Device Capabilities	Captured Slot Power Limit Value Captured Slot Power Value	Contains the value from the most recent Set Slot Power message received on this Port. For an MR Link, Set Slot Power messages on any VH will update this field. Default value is 0 (also see Section 4.3.7.3).
Device Capabilities	Function Level Reset Capability	Must be 0b.
Device Control	Correctable Error Reporting Enable Non-Fatal Error Reporting Enable Fatal Error Reporting Enable Unsupported Request Reporting Enable	Not Implemented. Errors are raised and controlled within a VH.
Device Control	Enable Relaxed Ordering	Not Implemented. Controlled within a VH.
Device Control	Max_Payload_Size	Not Implemented. Controlled within a VH.
Device Control	Extended Tag Field Enable	Not Implemented. Controlled within a VH.
Device Control	Phantom Function Enable	Must be 0b.
Device Control	Enable No Snoop	Not Implemented. Controlled within a VH.
Device Control	Max_Read_Request_Size	Not Implemented. Controlled within a VH.
Device Status	Correctable Error Detected Non-Fatal Error Detected Fatal Error Detected Unsupported Request Detected Unsupported Request Detected	Not Implemented. Errors are raised and controlled within a VH.
Device Status	Transactions Pending	Not Implemented. Meaningful only within a VH.
Link Capabilities	Surprise Down Error Reporting Capable Data Link Layer Active Reporting Capable Link Bandwidth Notification Capable	Must be 1b.
Link Capabilities	Port Number	Not Implemented. In MR-IOV, Port Number is the index into the Port table. Switch PCIe Port Number that is visible in the VH comes from VS Bridge Table (see Sections 4.3.6.3 and 4.3.7.3).
Link Control	Retrain Link	Implemented for all Ports, not just Downstream Ports.
Link Control	Link Disable	If Link Direction Supported indicates Crosslinks are supported, this field is implemented for both Upstream and Downstream Ports and is not affected by Port Direction Control.

Register	Field(s)	Attributes
Slot Capabilities	Attention Button Present Power Controller Present MRL Sensor Present Attention Indicator Present Power Indicator Present Hot-Plug Surprise Hot-Plug Capable Electromechanical Interlock Present	HwInit. Reflects Physical Hot-Plug capabilities.
Slot Capabilities	Slot Power Limit Value Slot Power Limit Scale	Hwinit. Value indicates what will be sent in any Set Slot Power message sent out this Port. Writing this register has no effect (Set_Slot_Power_Limit are sent using the PCIe Capability, see Section 4.3.7.3).
Slot Control	All fields	Reflect Physical Slot/Hot-Plug controls. Software Notification causes the Physical Hot-Plug Interrupt Pending bit in the Port Status register to be Set.
Device Capabilities 2	Completion Timeout Ranges Supported Completion Timeout Disable Supported ARI Forwarding Supported	Not Implemented. Implemented within a VH.
Device Control 2	Completion Timeout value Completion Timeout Disable ARI Forwarding Enable	Not Implemented. Implemented within a VH.

4.3.3.13. Port Interrupt Status Bitmap (minus 20h)

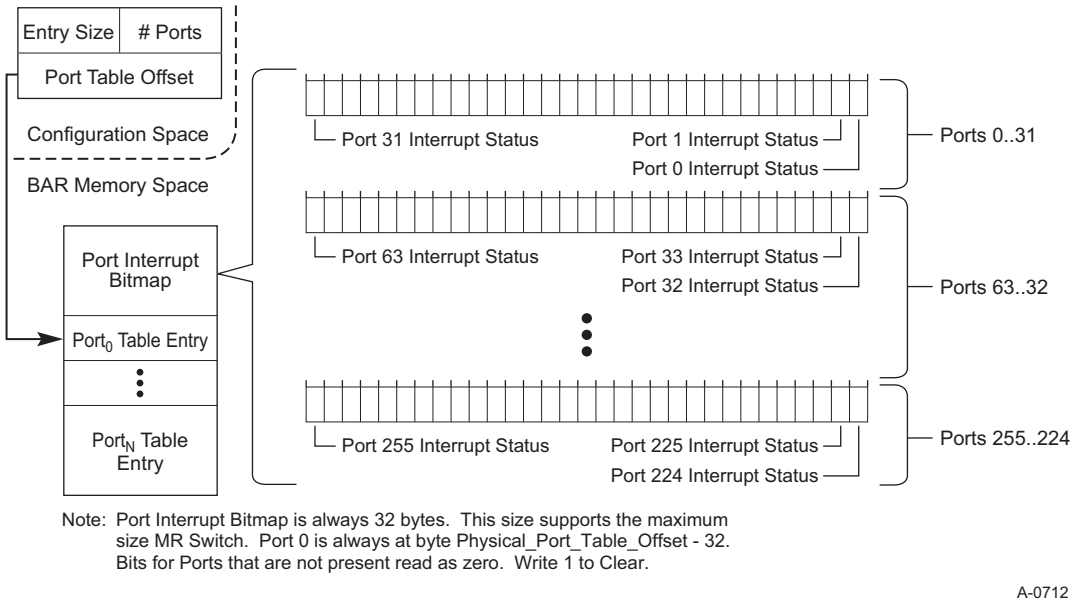

Figure 4-9: Port Interrupt Status Bitmap

The Port Interrupt Status bitmap precedes the Port Table. It is always 32 bytes (supporting Switches with a maximum of 256 Ports).

Bits in this table are Read Only. A bit is Set to indicate the Port has an interrupt pending and Clear otherwise. These Interrupt Status bits are cleared either by clearing the appropriate Port Interrupt Pending bit or by masking the interrupt using the Port Interrupt Enable.

An MSI Interrupt is requested on any zero to one transition of any of these bits.

Bits corresponding to Ports that are not Present are Read Only Zero. This bitmap is Read Only Zero unless the associated VS is Authorized.

4.3.4. Switch VL Arbitration Table

Switch and Device VL Arbitration tables are identical. See Section 4.3.7.3 for details.

4.3.5. Switch VS Table

The VS Table contains up to 256 VS Table Entries. This table is located in memory starting at a location determined by the VS Table Offset field in the MR-IOV Capability. This table is Read Only Zero unless the associated VS is Authorized.

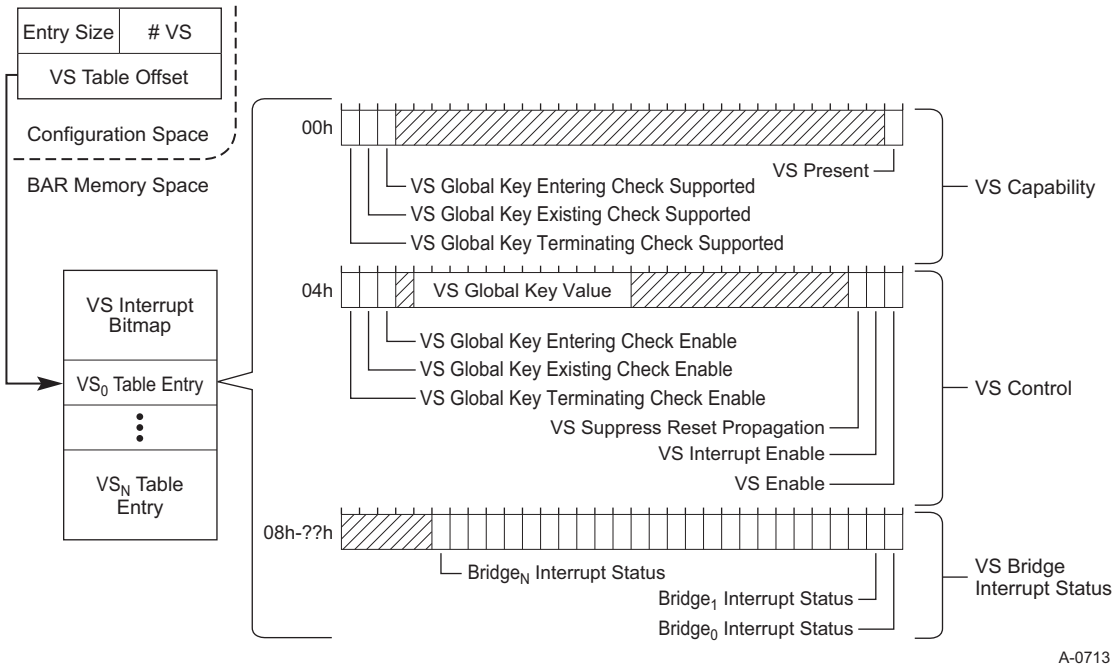

Figure 4-10: VS Table

4.3.5.1. VS Capability and Status (00h)

Table 4-53: Switch VS Capability and Status

Bit Location	Register Description	Attributes
0	**VS Present** – Read only bit indicating that the VS is implemented. When VS Enable is zero, this bit may be controlled with a Vendor Specific mechanism to allow for flexible silicon implementation.	RO
28:1	**Reserved**	RO
29	**VS Global Key Entering Check Supported** – Indicates the Switch supports the optional Global Key Entering Check. If supported, this check must be implemented in every VS.	RO
30	**VS Global Key Exiting Check Supported** – Indicates the Switch supports the optional Global Key Exiting Check. If supported, this check must be implemented in every VS.	RO

Bit Location	Register Description	Attributes
31	**VS Global Key Terminating Check Supported** – Indicates the Switch supports the Global Key Terminating Check. This support is mandatory so this bit must be 1b.	RO

4.3.5.2. VS Control (04h)

Table 4-54: Switch VS Capability and Status

Bit Location	Register Description	Attributes
0	**VS Enable** – Indicates that MR-PCIM is using this VS. When Set, the capabilities of the VS may not change. When Clear, Vendor Specific mechanisms may change the capabilities offered by this VS. If the Switch does not support changing VS capabilities, this bit may be read only with the value 1b. The default value of this field is vendor specific.	RW
1	**VS Interrupt Enable** – If Set, the VS Interrupt Summary bit corresponding to this VS will be affected when any of the Bridge N Interrupt Status bits in this VS are Set. If Clear, the Bridge N Interrupt Status bits have no effect on the corresponding VS Interrupt Summary bit. Default is 0b.	RW
2	**VS Suppress Reset Propagation** – If Set, the automatic sending of Hot Reset or Reset DLLPs downstream is suppressed. If Clear, DL_Down, Hot Reset, and/or Reset DLLPs received on the upstream Bridge of this VS cause downstream Bridges to send Hot Reset or Reset DLLPs. Suppressing Reset Propagation can be used to ensure that a failure in the management VH does not prematurely reset the entire MR Topology. Suppressing Reset Propagation does not affect TLP discarding. TLPs destined to or from a Bridge that is down or in Reset will be discarded even though this bit causes Reset at the associated Link to be suppressed. Changing this bit can cause a Link to enter or exit reset. This bit does not affect operation of and will not suppress resets caused by either the Secondary Bus Reset bit in the Bridge's Type 1 Configuration Header or the Virtual Force Reset bit in the VS Bridge Table. The default value of this field is vendor specific.	RW
15:3	**Reserved**	RO
27:16	**VS Global Key Value** – Expected Global Key Value for TLPs associated with the VS. This value is inserted in TLPs originated by the VS. This value is inserted in TLPs entering the VS from a non-MR Enabled Link. If enabled, Global Keys for TLPs associated with the VS are checked against this value. The default value of this field is 000h.	RW
28	**Reserved**	RO

Bit Location	Register Description	Attributes
31:29	**VS Global Key Entering Check Enable** – Enables checking of the Global Key in the TLP Prefix against the Global Key Value for TLPs associated with the VS. The Entering Check validates forwarded TLPs as they are received by the Switch. This error is reported in the Port where the TLP entered the Switch. If VS Global Key Entering Check Supported is Clear, this bit is Read Only Zero. Default is 0b.	RW
30	**VS Global Key Exiting Check Enable** – Enables checking of the Global Key in the TLP Prefix against the Global Key Value for TLPs associated with the VS. The Exiting Check validates forwarded TLPs as they are transmitted by the Switch. This error is reported in the Port where the TLP exited the Switch. If VS Global Key Exiting Check Supported is Clear, this bit is read only zero. Default is 0b.	RW
31	**VS Global Key Terminating Check Enable** – Enables checking of the Global Key in the TLP Prefix against the Global Key Value for TLPs associated with the VS. The Terminating Check validates TLPs addressing the VS and TLPs being forwarded to a Base PCIe Link. This error is reported in the Port where the TLP entered the Switch. Default is 0b.	RW

4.3.5.3. VS Bridge Interrupt Status (08h to ??h)

This field contains one bit per P2P Bridge in the VS. Bit 0 of the first DWORD corresponds to VS Bridge Table Entry 0 (i.e., the Upstream Bridge). Bit 1 corresponds to the VS Bridge Table Entry 1. This field contains INT((Num_Bridge_Table_Entries + 31)/32) DWORDs.

Bits in this field are Set if any of the following bits are Set in the VS Bridge Table.

- VC Status Changed
- Attention Indicator State Changed
- Power Indicator State Changed
- Power Controller State Changed
- PME Turn Off State Changed

Bits in this field are Cleared by clearing the associated "Changed" bits.

If VS Interrupt Enable is Set and any of the bits in this field are Set, the corresponding VS Interrupt Summary bitmap bit is Set.

4.3.5.4. VS Interrupt Status Bitmap (Minus 20h)

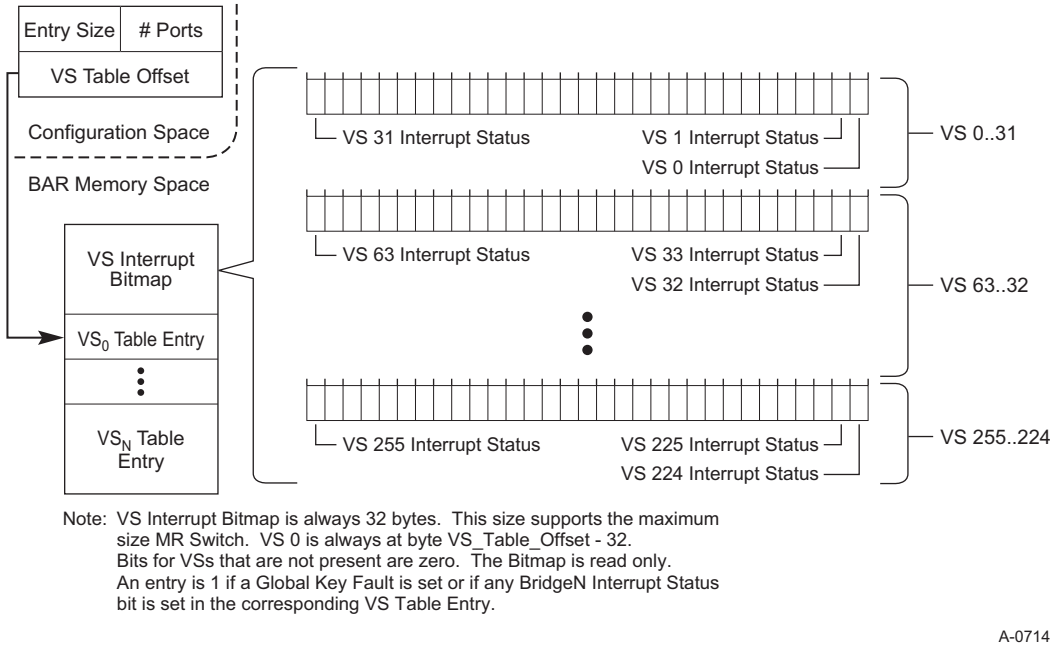

Figure 4-11: VS Interrupt Status Bitmap

The VS Interrupt Status bitmap precedes the VS Table. It is always 32 bytes (supporting Switches with a maximum of 256 Virtual Switches).

Bits in this table are Read Only. A bit is Set to indicate the VS has an interrupt pending and Clear otherwise. These Interrupt Status bits are cleared by clearing the VS BridgeN Interrupt bit Pending bit using the VS Bridge Table or by masking the interrupt using the VS Interrupt Enable.

An MSI Interrupt is requested on any zero to one transition of any of these bits.

Bits corresponding to Virtual Switches that are not Present are Read Only Zero. This bitmap is Read Only Zero unless the associated VS is Authorized.

4.3.6. Switch VS Bridge Table

The VS Bridge Table contains up to 256 Bridge Table Entries. This table is located in memory starting at a location determined by the VS Bridge Table Offset field in the MR-IOV Capability. This table is Read Only Zero unless the associated VS is Authorized.

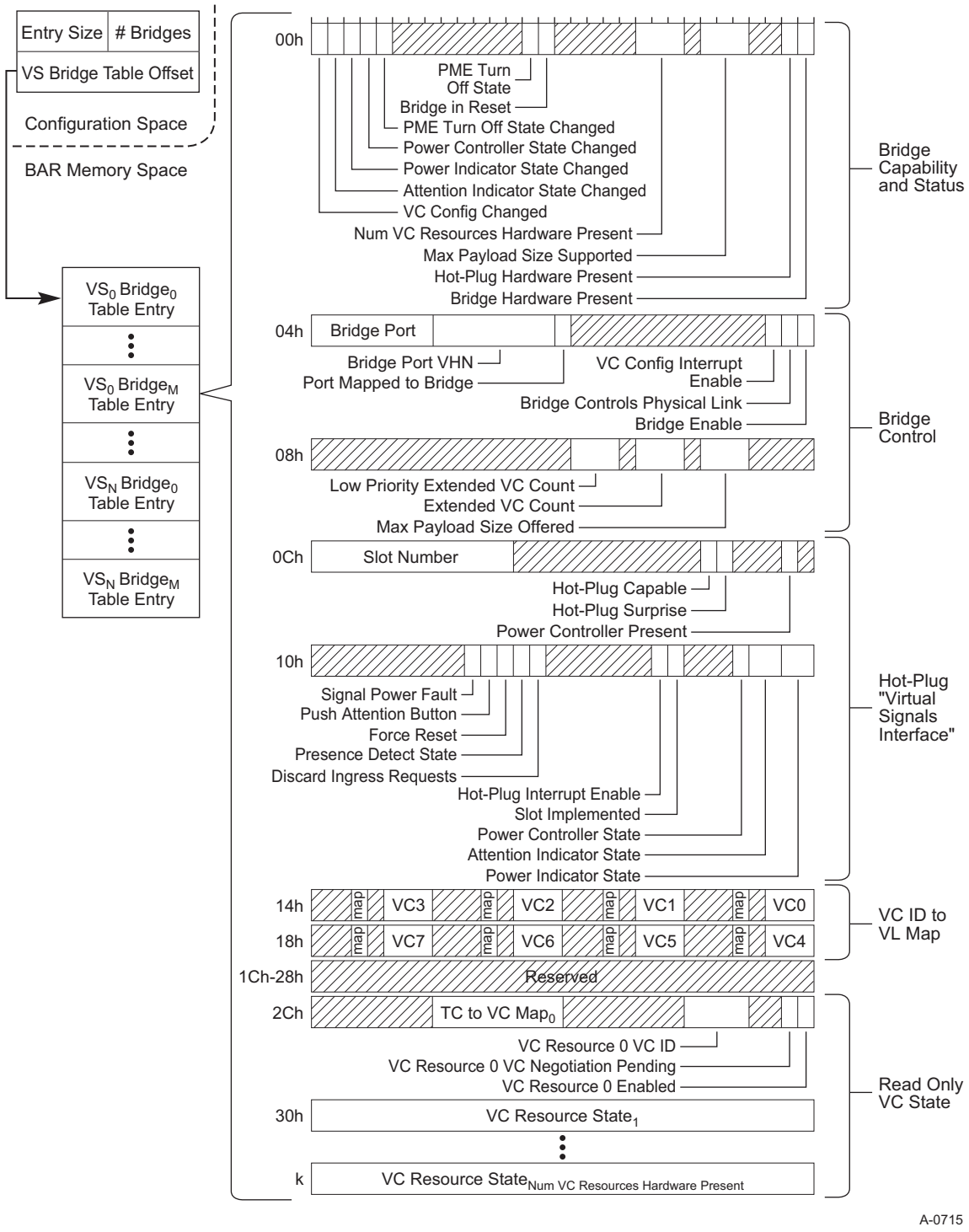

Figure 4-12: VS Bridge Table

VS Bridge Table Entry 0 corresponds to the upstream P2P Bridge of the VS. This bridge is always present.

VS Bridge Table Entry 1 corresponds to the downstream P2P Bridge located at Device 0, Function 0 on the VS internal bus. VS Bridge Table Entry *N* (1≤*N*≤32) corresponds to the downstream P2P

Bridge located at Device *N-1*, Function 0 on the VS Internal bus. VS Bridge Table Entries above 32 are used for P2P Bridges located at non-zero Function numbers as shown in Table 4-55.

Table 4-55: VS Bridge Table Entries

VS Bridge Table Entry *N*	Device	Function
1..32	*N*-1	0
33..64	*N*-33	1
65..96	*N*-65	2
97..128	*N*-97	3
129..160	*N*-129	4
161..192	*N*-161	5
193..224	*N*-193	6
225..256	*N*-225	7

4.3.6.1. VS Bridge Capability and Status (00h)

Table 4-56: Switch VS Bridge Capability and Status

Bit Location	Register Description	Attributes
0	**Bridge Hardware Present** – Set if the hardware supports a bridge in this slot. The VS Bridge Table has the same number of Table Entries associated with each VS. This bit allows the table to be sparse and to be populated in a Vendor Specific manner. This bit is not allowed to change if VS Enabled is Set. The Upstream Bridge is always present in any enabled VS. When VS Enable is Clear, Vendor Specific mechanisms may change the value of this field.	RO
1	**Hot-Plug Hardware Present** – This field indicates whether the P2P Bridge supports Hot-Plug. If Clear, Hot-Plug hardware is not present and the remainder of the "signals interface" bits are read only zero. This bit is always Clear for the upstream Bridge of a VS. When either VS Enable or Port Mapped to Bridge are Clear, Vendor Specific mechanisms may change the value of this field.	RO
3:2	**Reserved**	RO
6:4	**Max Payload Size Supported** – Returns the maximum Payload Size supported by this bridge. When VS Enable is Clear, Vendor Specific mechanisms may change the value of this field.	RO
7	**Reserved**	RO
10:8	**Num VC Resources Hardware Present** – Indicates the index of the last VC Resource array structure implemented in the Type 1 Configuration header associated with this Bridge in the VH. The value 0 indicates one VC Resource is provided. The value 7 indicates that all 8 VC Resources are provided. This value indicates the number that the hardware implements. MR-PCIM software may offer a lower number to the VH by setting Extended VC Count. When either VS Enable or Port Mapped to Bridge are Clear, Vendor Specific mechanisms may change the value of this field.	RO
15:11	**Reserved**	RO
16	**Link in Reset** – if Set indicates that the link mapped to this bridge is in the Reset state. For an upstream bridge this bit is Set if the link is an MR Link and the reset state machine is in state DS VH Down (see Section 2.3.3.2) or if the link is a PCIe Link that is in Hot Reset. For a downstream bridge, this bit is Set if the link is an MR Link and the reset state machine is in state US VH Down (see Section 2.3.3.1) or if the link is a PCIe link that is in Hot Reset. This field is Set if Port Mapped to Bridge is Clear.	RO

Bit Location	Register Description	Attributes
17	**PME Turn Off State** – Set when this bridge completes the PME Turn Off Handshake. Cleared when this bridge receives any TLP other than PME_Turn_Off or PME_TO_Ack or when this bridge enters Reset. The PME Turn Off handshake completes when an Upstream bridge sends or a Downstream Bridge receives a PME_TO_Ack message.	RO
26:18	**Reserved**	RO
27	**PME Turn Off State Changed** – This field indicates to MR-PCIM that the PME Turn Off State bit has changed state. Default is 0b.	RW1C
28	**Power Controller State Changed** – This field indicates to MR-PCIM that the Power Controller State has been changed by software in the VH. Default is 0b.	RW1C
29	**Power Indicator State Changed** – This field indicates to MR-PCIM that the Power Indicator State has been changed by software in the VH. Default is 0b.	RW1C
30	**Attention Indicator State Changed** – This field indicates to MR-PCIM that the Attention Indicator State has been changed by software in the VH. Default is 0b.	RW1C
31	**VC Config Changed** – Indicates that software running in the VH changed the VC Configuration of some VC Resource associated with this Bridge. This bit is Set if a VC Resource is enabled or disabled. This bit is also Set if the VC ID or TC to VC Map is changed while the VC is Enabled. Default is 0b.	RW1C

4.3.6.2. VS Bridge Control (04h and 08h)

Table 4-57: Switch VS Bridge Control 1

Bit Location	Register Description	Attributes

Bit Location	Register Description	Attributes
0	**Bridge Enable** – If set, the associated P2P Bridge is visible in the associated VS and the associated Type 1 Configuration header is accessible. If clear, the associated P2P Bridge is disabled. Accesses by software in the VS to Memory and Configuration Space of the Bridge as well as Memory, Configuration, and I/O Space of Functions below the Bridge will return UR. Reset will not propagate through this Bridge. Clearing Bridge Enable for an Upstream VS Bridge Table entry blocks access to bridges below the upstream bridge and thus has the effect of hiding the entire VS from the view of software running in the VH. Enabling a VS is a two step process. First set VS Enable. This ensures that Vendor Specific mechanisms will not alter the Bridge Hardware Present bits during the rest of this process. Then configure and enable specific bridges in the VS starting with downstream bridges and finishing with the upstream bridge. If Bridge Hardware Present is Clear, this field is Read Only Zero. The default value of this field is Vendor Specific.	RW

Bit Location	Register Description	Attributes
1	**Bridge Controls Physical Link** – If Set and this Bridge is mapped to some Port's VH0, the fields in this Bridge's Type 1 header control the Physical Link and are not Virtual. Otherwise, the fields in this Bridge's Type 1 header are Virtual. The fields involved are all fields from the PCI Express Capability and selected additional fields from the Type 1 header (see Section 4.3.3.8 for the list of fields affected). If this bit is Set, the registers in the Type 1 header are the same as the registers in the Switch Port Table Entry corresponding to the Port mapped to this Bridge. Changes to Type 1 header bits for the bridge where this bit is Set affects the Type 1 header and also affects the equivalent register in the Port Table. Similarly, changes in the Port Table registers are visible in the Type 1 header corresponding to the Bridge where this bit is Set. By setting this bit, MR-PCIM is ceding control of the physical Link to software running in the associated VH. This is necessary when a Switch is being used as a Base PCIe Switch. This may also be useful for Base PCIe links attached to MR Switches. This Bridge is mapped to VH0 of some Port if (1) the Port Mapped to Bridge bit is Set, (2) the Bridge VHN field is 0, and (3) the Bridge Port field contains a valid Port number. The results are undefined if an Upstream Bridge has this bit Set and the Port's Link Direction Control is not Upstream or if a Downstream Bridge has this bit Set and the Port's Link Direction Control is not Downstream. If this Bridge is not mapped to VH 0 of some Port, this bit is either Read Only Zero or Read/Write with the value ignored. If this bit is Set in an upstream port, then power management of the associated link follows rules defined in the *PCI Express Base Specification* (see Send PME_Enter_L23 DLLP bit in section 4.3.3.2). The default value of this field is Vendor Specific.	RW
2	**VC Config Interrupt Enable** – If Set, the VC Config Changed bit can trigger an interrupt. If Clear, VC Config Changed will not trigger an interrupt. Default is 0b.	RW
3	**PME Turn Off State Change Interrupt Enable** – If Set, the PME Turn Off State Changed bit can trigger an interrupt. If Clear, PME Turn Off State Changed will not trigger an interrupt. Default is 0b.	RW

Bit Location	Register Description	Attributes
14:4	**Reserved**	RO
15	**Port Mapped to Bridge** – If Clear, no Port is mapped to this Bridge. The Bridge VHN and Bridge Port fields are not used. For Downstream Bridges, software operating in the VH sees the Data Link Layer Link Active bit Clear and Link is in the virtual DL_Inactive state.[14] If Set, a Port is mapped to this Bridge. The Bridge VHN and Bridge Port fields are used. The virtual Data Link Layer Link Active bit and the virtual Link state track the physical Link state of the mapped Port. Changes to the Virtual Data Link Layer Link Active bit, either through changes to this bit or changes to the physical Link state, cause a Data Link State Change event in the virtual Hot-Plug controller. TLPs arriving at a {Port, VH} that does not have some VS Bridge mapped to it are discarded and an MR Uncorrectable Non-Fatal TLP Error is signaled. The default value of this field is Vendor Specific. If Bridge Enable is Clear, this field is Read Only Zero. For Upstream Bridges, changing this bit while Link in Reset is Clear is undefined.	RW
23:16	**Bridge VHN** – This field indicates the Port VHN associated with this P2P Bridge. This value must be less or equal to than value of NumVH in the Port contained in Bridge Port. Hardware ignores the value of this field if the Port Mapped to Bridge bit is Clear. The default value of this field is Vendor Specific. This field is Read Only if Port Mapped to Bridge is Set. To change a mapping, software must first clear Port Mapped to Bridge.	RW
31:24	**Bridge Port** – This field indicates the Port associated with this P2P Bridge. This value must correspond to an enabled Port. Hardware ignores the value of this field if the Port Mapped to Bridge bit is Clear. The default value of this field is Vendor Specific. This field is Read Only if Port Mapped to Bridge is Set. To change a mapping, software must first clear Port Mapped to Bridge.	RW

Behavior is undefined if a Bridge Port/Bridge VHN combination is simultaneously mapped into more than one VS Bridge Table entry.

Table 4-58: Switch VS Bridge Control 2

Bit Location	Register Description	Attributes
3:0	**Reserved**	RO

[14] For Upstream Bridges attached to MR Links, MR-PCIM software can use the Port Mapped to Bridge field in the Downstream Bridge of the Upstream Switch to cause this behavior. For Upstream Bridges attached to PCIe Links, MR-PCIM can disable the link using the Port Table to cause this behavior.

Bit Location	Register Description	Attributes
6:4	**Max Payload Size Offered** – Indicates Maximum Payload Size offered in the Type 1 header. This value must be less than or equal to the Max Payload Size Supported value. Default value for this field is Vendor Specific.	RW
7	**Reserved**	RO
10:8	**Extended VC Count** – MR-PCIM must configure this value with the number of VC Resources that are offered to software operating in the VH. This value will be available as Extended VC Count in the VC Capability of the associated Type 1 Configuration header. This value must be less than or equal to Num VC Resources Hardware Present. Changing this value while Bridge Enable is set is undefined. The default value of this field is Vendor Specific.	RW
11	**Reserved**	RO
14:12	**Low Priority Extended VC Count** – MR-PCIM must configure this value with the value to be provided to software operating in the VH. This value has no hardware effect. This field's purpose is to inform software running in the VH of the relative priority of certain VCs. This value must be less than or equal to the value set for Extended VC Count. Changing this value while Bridge Enable is set is undefined. The default value of this field is Vendor Specific.	RW
31:15	**Reserved**	RO

4.3.6.3. Hot-Plug Virtual Signals Interface (0Ch and 10h)

These bits form the "Virtual Signals Interface" of the Virtual Hot-Plug controller. These registers allow MR-PCIM to indicate to software what Hot-Plug features are supported and to control those features.

See Chapter 6 for additional details.

Virtual Hot-Plug controller hardware is optional. Presence of hardware is indicated by the Hot Plug Hardware Present bit. If no hardware is present, the Slot Implemented bit is Read Only zero and some of the bits in this section are Undefined. Virtual Hot-Plug hardware is only present in Downstream Ports.

If the Bridge Controls Physical Link bit is Set, the Virtual Hot-Plug Signals Interface registers are not used and their content is Undefined.

Table 4-59: Virtual Hot-Plug Signals Interface 1

Bit Location	Register Description	Attributes
0	**Reserved**	RO

Bit Location	Register Description	Attributes
1	**Virtual Power Controller Present** – This field indicates to software in the VH that the virtual slot has a Power Controller. This is visible to software in the VH through the Power Controller Present field of the PCI Express Capabilities. If Set, a Virtual Power Controller exists in the VH. When software operating in the VH turns off power to the virtual slot using the Virtual Hot-Plug Power Controller, a per-VH Reset is automatically triggered to clean out state in downstream hardware. The default value of this field is Vendor Specific. If Hot-Plug Hardware Present is 0b, this field is undefined.	RW
4:2	**Reserved**	RO
5	**Virtual Hot Plug Surprise** – This field indicates to software in the VH that the slot supports Surprise Hot-Plug events. Writing this field changes the Slot Capabilities field of the PCI Express Capabilities. The default value of this field is Vendor Specific. If Hot-Plug Hardware Present is 0b, this field is undefined.	RW
6	**Virtual Hot Plug Capable** – This field indicates to software in the VH that the slot is Hot-Plug Capable. Writing this field changes the Slot Capabilities field of the PCI Express Capabilities. The default value of this field is Vendor Specific. If Hot-Plug Hardware Present is 0b, this field is undefined.	RW
18:7	**Reserved**	RO
31:19	**Virtual Slot Number** – This field indicates the chassis slot number. It is echoed to software in the VH via the PCI Express Capabilities field. The default value of this field is Vendor Specific. If Hot-Plug Hardware Present is 0b, this field is undefined.	RW

Table 4-60: Hot-Plug Signals Interface 2

Bit Location	Register Description	Attributes
1:0	**Virtual Power Indicator State** – This field indicates the state of the Power Indicator as set by software in the VH. This corresponds to the Power Indicator Control field of the PCI Express Capability Slot Control register. Changes to this value set the Power Indicator Changed bit. If Hot-Plug Hardware Present is 0b, this field is undefined.	RO
3:2	**Virtual Attention Indicator State** – This field indicates the state of the Attention Indicator as set by software in the VH. This corresponds to the Attention Indicator Control field of PCI Express Capability Slot Control register. Changes to this value set the Attention Indicator Changed bit. If Hot-Plug Hardware Present is 0b, this field is undefined.	RO

Bit Location	Register Description	Attributes
4	**Virtual Power Controller State** –This field indicates the state of the Power Controller as set by software in the VH. This corresponds to the Power Controller Control bit of the Slot Control field of the PCI Express Capabilities. Changes to this value set the Power Controller Changed bit. If Hot-Plug Hardware Present is 0b, this field is undefined.	RO
7:5	**Reserved**	RO
8	**Virtual Slot Implemented** – This field indicates to software in the VH whether the P2P Bridge contains Hot-Plug support. Writing this field changes the Slot Implemented bit in the PCI Express Capabilities field. Default is 0b. If Hot-Plug Hardware Present is 0b, this field is Read Only Zero.	RW
9	**Hot Plug Signals Interrupt Enable** – If Set, the Power Controller Changed, Power Indicator Changed, and Attention Indicator Changed bits can trigger an interrupt. If Clear, these fields will not trigger an interrupt. Default is 0b. If Hot-Plug Hardware Present is 0b, this field is Read Only Zero.	RW
15:10	**Reserved**	RO
16	**Virtual Discard Ingress Request** – When this field is Set, the switch discards and returns credits for all Posted and Non-Posted TLPs that enter the switch through this Bridge. This field does not affect processing of Completion TLPs. This field is always present, even if Hot-Plug Hardware Present is 0b. Note: This field is used in conjunction with VS Suppress Reset Propagation (see Section 4.3.5.2) to support failover of MR-PCIM. Default is 0b.	RW
17	**Virtual Presence Detect State** – This field indicates that the virtual slot has a card in it. Defined Encodings are: 0b Virtual Slot Empty 1b Card Present in Virtual Slot The value of this field affects both the Presence Detect State and the Presence Detect Changed fields in the PCI Express Capability Slot Status register. The Presence Detect State field contains the same value as this field. The Presence Detect Changed field is Set whenever this field changes state. Default is 0b. If Hot-Plug Hardware Present is 0b, this field is undefined.	RW

Bit Location	Register Description	Attributes
18	**Virtual Force Reset** – For downstream Ports, when this field is Set, the Switch causes the Port/Port VHN mapped to this Bridge to enter Reset. The effect is the same as if software running the VH set the Secondary Bus Reset bit in the associated Type 1 header. For upstream Ports, this field is Read Only Zero. For downstream Ports, this field is always present, even if Hot-Plug Hardware Present is 0b. Operation of this bit is not affected by the VS Suppress Reset Propagation bit. Default is 0b.	RW
19	**Push Virtual Attention Button** – Writing 1 to this field Sets the Attention Button Pressed bit in the PCI Express Capability Slot Status register. This simulates the press of the virtual Attention Button in the Virtual Hot-Plug Controller. Writing 0 to this field has no effect. If Hot-Plug Hardware Present is 0b, writing to this field has no effect. This field reads as Zero.	Write 1 to Trigger, Read Zero
20	**Signal Virtual Power Fault** – Writing 1 to this field Sets the Power Fault Detected bit in the PCI Express Capability Slot Status register. This simulates a Power Fault condition in the Virtual Hot-Plug Controller. Writing 0 to this field has no effect. If Hot-Plug Hardware Present is 0b, writing to this field has no effect. This field reads as Zero.	Write 1 to Trigger, Read Zero
31:21	**Reserved**	RO

4.3.6.4. VS Bridge VC ID to VL Map (14h and 18h)

These fields contain the Virtual Link to be used for traffic out of this P2P Bridge for the indicated VC. VC to VL mapping is not needed and these fields are read only zero if the MaxVL value is zero for all Ports of the Switch.

Software may not map multiple VCs to the same VL. Specifically, within a single VS Bridge Table entry, behavior is undefined if multiple enabled VL Map entries contain the same map value.

Table 4-61: Switch VS Bridge VC ID to VL Map 1

Bit Location	Register Description	Attributes
2:0	**VC0 VL Map** – Indicates the VL number used for VS traffic labeled VC0 transmitted via this VS Bridge. This field is Read Only when VC0 VL Map Enable is Set. The default value of this field is Vendor Specific.	RW
3	**Reserved**	RO
4	**VC0 VL Map Enable** – Indicates that the VC0 VL Map field contains a valid VL number. The default value of this field is Vendor Specific. Hardware behavior on the 1b to 0b transition of this bit is undefined.	RW
7:5	**Reserved**	RO
10:8	**VC1 VL Map** – Indicates the VL number used by VS traffic labeled VC1 transmitted via this VS Bridge. This field is Read Only Zero if Num VC Resources Hardware Present is 0. This field is Read Only when VC1 VL Map Enable is Set. The default value of this field is Vendor Specific.	RW
11	**Reserved**	RO
12	**VC1 VL Map Enable** – Indicates that the VC1 VL Map field contains a valid VL number. This field is Read Only Zero if Num VC Resources Hardware Present is 0. The default value of this field is Vendor Specific. Hardware behavior on the 1b to 0b transition of this bit is undefined.	RW
15:13	**Reserved**	RO
18:16	**VC2 VL Map** – Indicates the VL number used by VS traffic labeled VC2 transmitted via this VS Bridge. This field is Read Only Zero if Num VC Resources Hardware Present is 0 or 1. This field is Read Only when VC2 VL Map Enable is Set. The default value of this field is Vendor Specific.	RW
19	**Reserved**	RO
20	**VC2 VL Map Enable** – Indicates that the VC2 VL Map field contains a valid VL number. This field is Read Only Zero if Num VC Resources Hardware Present is 0 or 1. The default value of this field is Vendor Specific. Hardware behavior on the 1b to 0b transition of this bit is undefined.	RW
23:21	**Reserved**	RO

Bit Location	Register Description	Attributes
26:24	**VC3 VL Map** – Indicates the VL number used by VS traffic labeled VC3 transmitted via this VS Bridge. This field is Read Only Zero if Num VC Resources Hardware Present is less than or equal to 2. This field is Read Only when VC3 VL Map Enable is Set. The default value of this field is Vendor Specific.	RW
27	**Reserved**	RO
28	**VC3 VL Map Enable** – Indicates that the VC3 VL Map field contains a valid VL number. This field is Read Only Zero if Num VC Resources Hardware Present is less than or equal to 2. The default value of this field is Vendor Specific. Hardware behavior on the 1b to 0b transition of this bit is undefined.	RW
31:29	**Reserved**	RO

Table 4-62: Switch VS Bridge VC ID to VL Map 2

Bit Location	Register Description	Attributes
2:0	**VC4 VL Map** – Indicates the VL number used by VS traffic labeled VC4 transmitted via this VS Bridge. This field is Read Only Zero if Num VC Resources Hardware Present is less than or equal to 3. This field is Read Only when VC4 VL Map Enable is Set. The default value of this field is Vendor Specific.	RW
3	**Reserved**	RO
4	**VC4 VL Map Enable** – Indicates that the VC4 VL Map field contains a valid VL number. The default value of this field is Vendor Specific. Hardware behavior on the 1b to 0b transition of this bit is undefined.	RW
7:5	**Reserved**	RO
10:8	**VC5 VL Map** – Indicates the VL number used by VS traffic labeled VC5 transmitted via this VS Bridge. This field is Read Only Zero if Num VC Resources Hardware Present is less than or equal to 4. This field is Read Only when VC5 VL Map Enable is Set. The default value of this field is Vendor Specific.	RW
11	**Reserved**	RO
12	**VC5 VL Map Enable** – Indicates that the VC5 VL Map field contains a valid VL number. This field is Read Only Zero if Num VC Resources Hardware Present is less than or equal to 4. The default value of this field is Vendor Specific. Hardware behavior on the 1b to 0b transition of this bit is undefined.	RW
15:13	**Reserved**	RO
18:16	**VC6 VL Map** – Indicates the VL number used by VS traffic labeled VC6 transmitted via this VS Bridge. This field is Read Only Zero if Num VC Resources Hardware Present is less than or equal to 5. This field is Read Only when VC6 VL Map Enable is Set. The default value of this field is Vendor Specific.	RW

Bit Location	Register Description	Attributes
19	**Reserved**	RO
20	**VC6 VL Map Enable** – Indicates that the VC6 VL Map field contains a valid VL number. This field is Read Only Zero if Num VC Resources Hardware Present is less than or equal to 5. The default value of this field is Vendor Specific. Hardware behavior on the 1b to 0b transition of this bit is undefined.	RW
23:21	**Reserved**	RO
26:24	**VC7 VL Map** – Indicates the VL number used by VS traffic labeled VC7 transmitted via this VS Bridge. This field is Read Only Zero if Num VC Resources Hardware Present is less than or equal to 6. This field is Read Only when VC7 VL Map Enable is Set. The default value of this field is Vendor Specific.	RW
27	**Reserved**	RO
28	**VC7 VL Map Enable** – Indicates that the VC7 VL Map field contains a valid VL number. This field is Read Only Zero if Num VC Resources Hardware Present is less than or equal to 6. The default value of this field is Vendor Specific. Hardware behavior on the 1b to 0b transition of this bit is undefined.	RW
31:29	**Reserved**	RO

4.3.6.5. VC Resource Fields (2Ch upto 48h)

These fields return data from the VC Capability of the associated Type 1 Configuration header. They allow MR-PCIM software to track the enabling and mapping of VCs with each VH.

VC Resource fields for resource numbers above Num VC Resource Hardware Present are not implemented and return 0 when read. VC Resource fields for resource numbers above Extended VC Count are not implemented and do not exist.

VC Resource State 0 is located at offset 2Ch; VC Resource State 1 is located at offset 30h; etc. Fields within VC Resource State are described in Table 4-63.

Table 4-63: VC Resource State

Bit Location	Register Description	Attributes
0	**VC Enabled** – This field tracks the VC Enabled bit set by software operating in the VH. Per the *PCI Express Base Specification*, VC Resource 0 is always enabled and thus VC Enabled for VC Resource 0 is always set.	RO
1	**VC Negotiation Pending** – This field tracks the VC Negotiation Pending bit view in the VH. This bit is Set when VC Enabled is Set and either no VL has been mapped to this VC (associated VL Map Enable bit is Clear) or Flow Control negotiation has not completed on the mapped VH and VL.	RO
3:1	**Reserved**	RO
6:4	**VC ID** – This field tracks the VC ID field set by software operating in the VH. Per the *PCI Express Base Specification*, VC ID for VC Resource 0 is always 0.	RO
15:7	**Reserved**	RO
23:16	**TC to VC Map** – This field tracks the TC to VC Map field set by software operating in the VH. Per the *PCI Express Base Specification*, bit 0 of this field is fixed and the remaining bits may be set by software. Also, per the *PCI Express Base Specification*, the default value of this field is FFh for the first VC Resource and is 00h for other VC Resources.	RO
31:24	**Reserved**	RO

4.3.7. Miscellaneous Switch Configuration Space Requirements

4.3.7.1. ARI Support

Alternate RID Interpretation (ARI) support must be provided in all downstream PCI-to-PCI bridges of MRA Switches. Specifically, the ARI Forwarding Supported bit located in the Device Capabilities 2 register must be set and the ARI Forwarding Enable bit located in the Device Control 2 register must be implemented.

4.3.7.2. BIST (Switch)

MR-IOV Switches shall not support BIST.

IMPLEMENTATION NOTE

No BIST Support

In a virtualized environment, BIST would also have to be virtualized. Since BIST is rarely used and not completely specified, it was decided to remove it from MR-IOV.

4.3.7.3. Switch PCIe Capability Fields

Behavior of certain fields in the PCI Express Capability changes due to virtualization. Unless indicated in Table 4-64, behavior is as specified in the *PCI Express Base Specification*.

Table 4-64: Switch PCIe Capability Fields

Register	Field(s)	Attributes: Bridge Control Physical is 0b	Attributes: Bridge Controls Physical is 1b
PCIe Capabilities	Slot Implemented	Set if Virtual Slot Implemented is Set in the VS Bridge Table entry.	Base
Device Capabilities	Max_Payload_Size Supported	Set from the Max_Payload_Size Offered value in the VS Bridge Table entry.	
Device Capabilities	Captured Slot Power Limit Value Captured Slot Power Value	If Upstream Bridge and Port Mapped to Bridge is Set, contains the same value as the Port Table fields. If Downstream Bridge or if Port Mapped to Bridge is Clear, contains 0.	
Device Capabilities	Function Level Reset Capability Role-Based Error Reporting	Must be 1b.	
Device Control	Max_Payload_Size	Fully Implemented. PCIe Base rules apply within each VH.	Base
Device Control	Auxiliary (AUX) Power PM Enable	Component is allowed to draw AUX power if at least one of the Functions, in any VH, has this bit set.	Base
Device Status	Aux Power Detected	Implemented	Base
Link Capabilities	Supported Link Speeds Maximum Link Width	If Port Mapped to Bridge is Set, returns the value associated with the Port. If Port Mapped to Bridge is Clear, returns the fastest and widest link supported by the Switch (i.e., the largest value that could occur).	Returns the value associated with the Port

Register	Field(s)	Attributes: Bridge Control Physical is 0b	Attributes: Bridge Controls Physical is 1b
Link Capabilities	ASPM Support	If Port Mapped to Bridge is Set, returns the value associated with the Port. If Port Mapped to Bridge is Clear, 11b if all Ports support L0s and L1, else returns 01b.	Returns the value associated with the Port
Link Capabilities	L0s Exit latency L1 Exit latency	If Port Mapped to Bridge is Set, returns the value associated with the Port. If Port Mapped to Bridge is Clear, returns the largest latency value for any Switch Port.	Returns the value associated with the Port
Link Capabilities	Clock Power Management	If Port Mapped to Bridge is Set, returns value associated with the Port. If Port Mapped to Bridge is Clear, returns 0b unless all Switch Ports support Clock Power Management.	Returns the value associated with the Port
Link Capabilities	Data Link Layer Link Active Reporting Capable	Set if Virtual Hot Plug Capable is Set	Returns the value associated with the Port
Link Capabilities	Surprise Down Error Reporting Capable Link Bandwidth Notification Capability	If Port Mapped to Bridge is Set, returns value associated with the Port. If Port Mapped to Bridge is Clear, returns 1b if any Switch Port supports the feature.	Returns the value associated with the Port
Link Capabilities	Port Number	Each Bridge in a VS has a unique Port Number value. Port Number values of different VS are not related.	In order to maintain uniqueness, returns the Port Number associated with the VS, not the Port.
Link Control	ASPM Control Common Clock Configuration Extended Sync Enable Clock Power Management Hardware Autonomous Width Disable	Implemented	Base

Register	Field(s)	Attributes: Bridge Control Physical is 0b	Attributes: Bridge Controls Physical is 1b
Link Control	ASPM Control Common Clock Configuration Extended Sync Enable Clock Power Management Hardware Autonomous Width Disable	As defined in the *PCI Express Base Specification*. If implemented, these are read/write fields but have no other effect.	Base
Link Control	Retrain Link	Read Only Zero, Writes are ignored unless Bridge Controls Physical is Set.	Base
Link Control	Link Disable	Implemented. If Bridge Controls Physical is Clear, has the same effect as setting Secondary Bus Reset.	Base
Link Control	Link Bandwidth Management Interrupt Enable Link Autonomous Bandwidth Interrupt Enable	Implemented	Base
Link Status	Current Link Speed Negotiated Link Width Link Training Slot Clock Configuration	Reflect the physical link values. Implemented in each VH	Base
Link Status	Link Training Data Link Layer Link Active	If Downstream Bridge and Port Mapped to Bridge is Set, returns value associated with the Port. Otherwise returns 0b.	Base
Link Status	Link Bandwidth Management Status Link Autonomous Bandwidth Status	Set based on events in the Port if this is a Downstream Bridge and if Port Mapped to Bridge is Set.	Base
Slot Capabilities	Hot-Plug Capable Attention Button Present Power Indicator Present Attention Indicator Present	Set if Virtual Hot Plug Capable is Set.	Base
Slot Capabilities	Power Controller Present	Set if the VS Bridge Table field Virtual Power Controller Present is Set.	Base

Register	Field(s)	Attributes: Bridge Control Physical is 0b	Attributes: Bridge Controls Physical is 1b
Slot Capabilities	MRL Sensor Present Electromechanical Interlock Present No Command Completed Support	Must be 0b.	Base
Slot Capabilities	Hot-Plug Surprise	Set if the VS Bridge Table field Virtual Hot Plug Surprise is Set.	Base
Slot Capabilities	Slot Power Limit Value Slot Power Limit Scale	If Downstream Bridge and Port Mapped to Bridge is Set, contains the same value as the Port Table fields and writes to this register cause the Port to send the Set_Slot_Power_Limit Message in the associated VH. If Upstream Bridge or Port Mapped to Bridge is Clear, contains 0.	Base
Slot Capabilities	Physical Slot Number	Contains the value of the VS Bridge table Virtual Slot Number field. Note that this value is always used since slot numbers must be unique within a VS even when the Ports for a VS have differing values for Bridge Controls Physical.	
Slot Control	Attention Button Pressed Enable Power Fault Detected Enable Presence Detect Changed Enable Hot-Plug Interrupt Enable Data Link Layer State Changed Enable	Read/Write	Base
Slot Control	MRL Sensor Changed Enable Command Completed Interrupt Enable Electromechanical Interlock Control	Hardwired to 0b	Base
Slot Control	Attention Indicator Control	Writing this field alters the VS Bridge Table field Virtual Attention Indicator State.	Base

Register	Field(s)	Attributes: Bridge Control Physical is 0b	Attributes: Bridge Controls Physical is 1b
Slot Control	Power Indicator Control	Writing this field alters the VS Bridge Table field Virtual Power Indicator State.	Base
Slot Control	Power Controller Control	Writing this field alters the VS Bridge Table field Virtual Power Controller State.	Base
Slot Status	Attention Button Pressed	Set when the VS Bridge Table field Signal Virtual Attention Button is written with 1b.	Base
Slot Status	Power Fault Detected	Set when the VS Bridge Table field Signal Virtual Power Fault is written with 1b.	Base
Slot Status	MRL Sensor Changed Command Completed MRL Sensor State Electromechanical Interlock Status	Hardwired to 0b	Base
Slot Status	Presence Detect Changed	Set when the VS Bridge Table field Virtual Presence Detect State is written to a different value.	Base
Slot Status	Presence Detect State	Contains the value of the VS Bridge Table field Virtual Presence Detect State.	Base
Slot Status	Data Link Layer State Changed	Base	Base
Device Capabilities 2	Completion Timeout Ranges Supported Completion Timeout Disable Supported	Optional. Not normally Implemented since MR-IOV Switches do not normally issue Non-Posted Transactions.	Base
Device Control 2	Completion Timeout value Completion Timeout Disable	Base	Base

Register	Field(s)	Attributes: Bridge Control Physical is 0b	Attributes: Bridge Controls Physical is 1b
Link Control 2	Target Link Speed Enter Compliance Hardware Autonomous Speed Disable Selectable De-emphasis Transmit Margin Enter Modified Compliance Compliance SOS Compliance De-emphasis	Read/Write registers, value is ignored.	Base
Link Status 2	Current De-emphasis Level	Contains the value associated with the Port.	Base

4.4. VL Arbitration Table

The VL Arbitration Table is optional. It is identical in structure to the VC arbitration table in PCI Express.

Switches and Devices use the same VL Arbitration Table structure. Switches and Devices differ in the location of the fields used to locate the VL Arbitration table and to configure the arbitration scheme used. For a Switch, fields in the Port Table are used for this purpose. For a Device, fields in the MR-IOV Capability are used instead.

The VL Arbitration Table is a read-write register array that is used to store the arbitration table for VL Arbitration. This register array is valid for all Functions when the selected VL Arbitration uses a WRR table. If it exists, the VL Arbitration Table is located by the VL Arbitration Table Offset field.

The VL Arbitration Table is a register array with fixed-size entries of 4 bits. Figure 4-13 depicts the table structure of an example VL Arbitration Table with 32 phases. Each 4-bit table entry corresponds to a phase within a WRR arbitration period. The definition of table entry is depicted in Table 4-65. The lower 3 bits (bits 0-2) contain the VL value, indicating that the corresponding phase within the WRR arbitration period is assigned to the Virtual Channel indicated by the VL (must be a valid VL that corresponds to an enabled VL).

The highest bit (bit 3) of the table entry is reserved. The length of the table depends on the selected VL Arbitration as shown in Table 4-66.

When the VL Arbitration Table is used by the default VL Arbitration method, the default values of the table entries must be all zero to ensure forward progress for the default VL (with VL of 0).

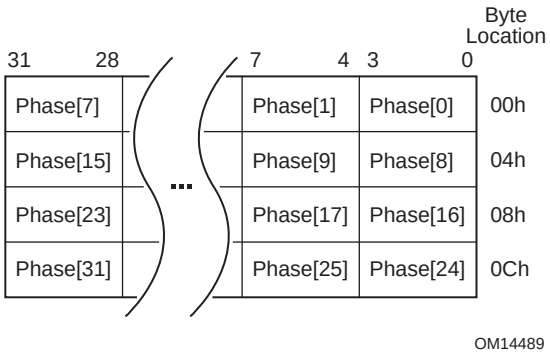

Figure 4-13: Example VL Arbitration Table with 32 Phases

Table 4-65: Definition of the 4-bit Entries in the VL Arbitration Table

Bit Location	Register Description	Attributes
2:0	**VL**	RW
3	**Reserved**	RsvdP

Table 4-66: Length of the VL Arbitration Table

VC Arbitration Select	VC Arbitration Table Length (in Number of Entries)
001b	32
010b	64
011b	128

4.5. Performance Monitoring and Statistics Collection

This section describes the registers and tables used to control the optional MR-IOV Performance Monitoring and Statistics Collection Capability.

Switch and Device usage is identical except where noted.

An overview of the registers and tables associated with this capability is shown in Figure 4-14. Detailed descriptions are provided in the following sections. The functional behavior is described in Section 8.3.

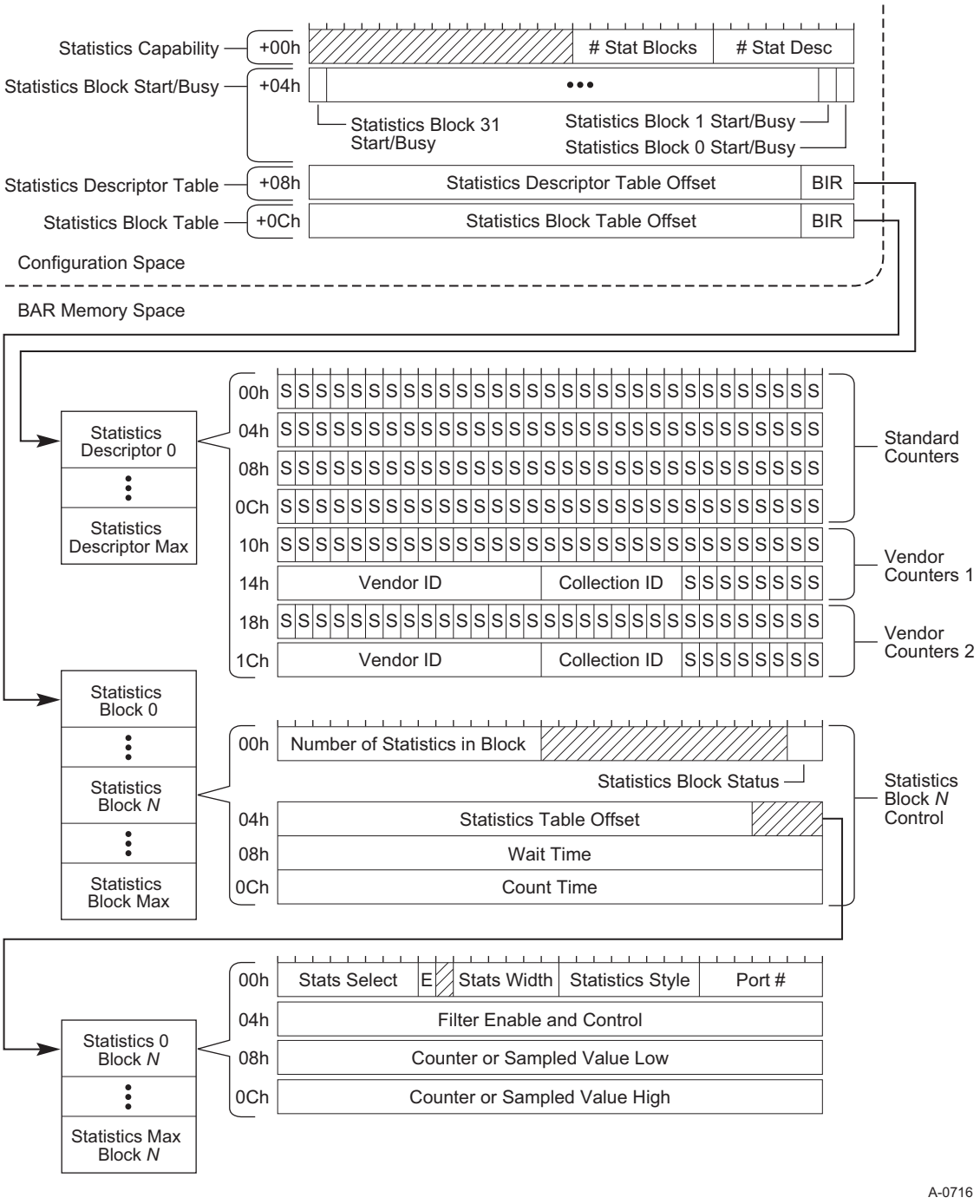

Figure 4-14: Performance Monitoring and Statistics Collection Tables

4.5.1. Configuration Space Fields

The following fields are located in the Switch and Device MR-IOV Capabilities. Starting offsets of these fields within these two capabilities are different.

In addition to the fields described below, each Capability contains Statistics Interrupt Status and Statistics Interrupt Enable bits.

For Switches, there is a single instance of the Performance Monitoring and Statistics Collection registers that is visible in all Authorized VSs. For all non-Authorized, VSs, the Performance Monitoring and Statistics Collection Capability is not implemented.

4.5.1.1. Statistics Capability (+00h)

These fields define the sizes of the Statistics Tables pointed to by the MR-IOV Capability.

If the Performance Monitoring and Statistics Collection Capability is not implemented, these fields are Read Only Zero.

Table 4-67: Statistics Table Sizes

Bit Location	Register Description	Attributes
7:0	**Number of Statistics Descriptors** – Indicates the number of entries in the Statistics Descriptor Table. Statistics support is optional for all components but strongly encouraged for MRA Switches. If not supported, this field is Zero.	RO
15:8	**Number of Statistics Blocks** – Indicates the number of entries in the Statistics Block Table. This value must be less than or equal to 32. Statistics support is optional for all components but strongly encouraged for MRA Switched. If not supported, this field is Zero.	RO
31:16	**Reserved**	RO

4.5.1.2. Statistics Block Start/Busy (+04h)

These fields contain a bit for each supported Statistics Block.

Table 4-68: Statistics Start/Busy

Bit Location	Register Description	Attributes
0	**Statistics Block 0 Start/Busy** – If idle, writes of 1b initiate the statistics collection processing for Statistics Block 0. The behavior of initiating statistics collection for a Statistics Block that is not idle is undefined. If not idle, writes of 0b terminate the statistics collection process for Statistics Block 0. Termination of the statistics collection process is not immediate; therefore, following termination of the statistics collection process, this field should be read to confirm that termination has completed and that the Statistics Block is idle. When read, indicates the busy status of the associated Statistics Block. The value 1b indicates the Statistics Block is busy (i.e., either waiting or counting). The value 0b indicates the Statistics Block is idle. If Number of Statistics Blocks is zero, this field is Read Only Zero.	RW
1	**Statistics Block 1 Start/Busy** – Controls Statistics Block 1. If Number of Statistics Block is 0 or 1, this field is Read Only Zero.	RW
2	**Statistics Block 2 Start/Busy** – Controls Statistics Block 2. If Number of Statistics Block is less than or equal to 2, this field is Read Only Zero.	RW
…	…	…
31	**Statistics Block 31 Start/Busy** – Controls Statistics Block 31. If Number of Statistics Block is less than or equal to 31, this field is Read Only Zero.	RW

4.5.1.3. Statistics Descriptor Table Offset (+08h)

If Number of Statistics Descriptors is zero, this table does not exist and this field is Read Only Zero.

Table 4-69: Statistics Descriptor Table Offset

Bit Location	Register Description	Attributes
2:0	**Statistics Descriptor Table BIR** – Indicates which one of a function's Base Address registers, located beginning at 10h in Configuration Space, is used to map the Function's Statistics Descriptor Table into Memory Space. **BIR Value Base Address register** 0 BAR0 10h 1 BAR1 14h 2 BAR2 18h (Device Only) 3 BAR3 1Ch (Device Only) 4 BAR4 20h (Device Only) 5 BAR5 24h (Device Only) 6..7 Reserved For a 64-bit Base Address register, the BIR indicates the lower DWORD. For Switch usage, the values 2..5 are Reserved as well.	RO
31:3	**Statistics Descriptor Table Offset** – Used as an offset from the address contained by one of the Function's Base Address registers to point to the base of the Statistics Descriptor Table. The lower 3 BIR bits are masked off (set to zero) by software to form a 32-bit offset that is QWORD aligned.	RO

4.5.1.4. Statistics Block Table Offset (+0Ch)

If Number of Statistics Blocks is zero, this table does not exist and this field is Read Only Zero.

Table 4-70: Statistics Block Table Offset

Bit Location	Register Description	Attributes
2:0	**Statistics Block Table BIR** – Indicates which one of a function's Base Address registers, located beginning at 10h in Configuration Space, is used to map the Function's Statistics Block Table into Memory Space. **BIR Value Base Address register** 0 BAR0 10h 1 BAR1 14h 2 BAR2 18h (Device Only) 3 BAR3 1Ch (Device Only) 4 BAR4 20h (Device Only) 5 BAR5 24h (Device Only) 6..7 Reserved For a 64-bit Base Address register, the BIR indicates the lower DWORD. For Switch usage, the values 2..5 are Reserved as well.	RO
31:3	**Statistics Block Table Offset** – Used as an offset from the address contained by one of the Function's Base Address registers to point to the base of the Statistics Block Table. The lower 3 BIR bits are masked off (set to zero) by software to form a 32-bit offset that is QWORD aligned.	RO

4.5.2. Statistics Descriptor Table

The Statistics Descriptor Table describes sets of statistics that may be recorded by Statistics Counters.

This table contains up to 256 entries. Each entry is 256 bits. The entire Statistics Descriptor Table is Read Only and constant.

A Descriptor Table Entry describes statistics recording capabilities. Each Statistics Counter contains the index of a Descriptor Table Entry that indicates the statistics recording capabilities associated with that counter.

Each Descriptor Table entry contains 208 Supported bits (S bits). If an S bit is 1b, then Statistics Counters that point to that Descriptor Table entry may be configured to record the statistic associated with the S bit. This configuration is done by setting the Statistics Select field of the counter to the bit number of the S bit.

S bits 127:0 represent statistics defined in this specification. S bits 167:128 and 231:192 represent Vendor Specific statistics.

Table 4-71: Statistics Descriptor Table Entry

Bit Location	Register Description	Attributes
127:0	**Standard S Bits** – Indicates S bits whose meaning is defined by this specification.	RO
167:128	**Group 1 Vendors Specific S Bits** – Indicates S bits whose meaning is defined by the Vendor listed in Group 1 Vendor ID further qualified by Group 1 Collection ID. If no Group 1 S bits are supported, this field is hardwired to zero.	RO
175:168	**Group 1 Collection ID** – Vendor Specific identifier that defines the meaning of bits 167:128. If no Group 1 S bits are supported, this field is hardwired to zero.	RO
191:176	**Group 1 Vendor ID** – Indicates the Vendor that defined the meaning of bits 175:128. If no Group 1 S bits are supported, this field is hardwired to zero.	RO
231:192	**Group 2 Vendors Specific S Bits** – Indicates S bits whose meaning is defined by the Vendor listed in Group 2 Vendor ID further qualified by Group 2 Collection ID. If no Group 2 S bits are supported, this field is hardwired to zero.	RO
239:232	**Group 2 Collection ID** – Vendor Specific identifier that defines the meaning of bits 231:192. If no Group 2 S bits are supported, this field is hardwired to zero.	RO
255:240	**Group 2 Vendor ID** – Indicates the Vendor that defined the meaning of bits 239:192. If no Group 2 S bits are supported, this field is hardwired to zero.	RO

The following nomenclature is used to describe counters:

- Standard statistics are designated CSEL[***n***] where ***n*** is in the range [0..127] (inclusive).
- Vendor Specific statistics are designated CSEL[Vendor ID, Collection ID, ***n***]. Vendor ID is assigned by the PCI SIG. Collection ID is a Vendor defined value used to select a set of S bit definitions. The value ***n*** is in the range [0..39] (inclusive) and the corresponding S bit number is ***n*** + 168 (if mapped using Group 1) or ***n*** + 192 (if mapped using Group 2). The meaning of a Vendor Specific statistic is not affected by whether it is mapped using Group 1 or Group 2.

This mechanism allows a single counter to support any mixture of standard events and vendor defined events from up to two sets of S bit definitions (from either the same or different Vendors).

4.5.2.1. Standard Statistics

Table 4-72 contains Standard Statistics defined by this specification. Items marked Sample correspond to sampled values while items marked Count correspond to counted values. Standard filters are defined in Section 4.5.2.2.

Table 4-72: Standard Statistics

S Bit Number	Description	Count/Sample	Applicable Filter(s)
0	**Transmitted TLPs** – Counts non-nullified TLPs transmitted by the Port. *Required to be implemented by at least two Statistics Counters per Port.*	Count	**Optional TLP Filters:** VH, VL, TLP Type
1	**Transmitted TLP DWORDs** – Counts DWORDs of non-nullified TLPs transmitted by the Port. This includes framing symbols and all bytes sent (i.e., STP to END inclusive). *Required to be implemented by at least two Statistics Counters per Port.*	Count	**Optional TLP Filters:** VH, VL, TLP Type
2	**Transmitted IDLE Symbols** – Counts the number of IDLE symbols transmitted by the Port. *Required to be implemented by at least two Statistics Counters per Port.*	Count	None
3	**Transmitted DLLPs** – Counts the number of DLLPs sent by this Port. *Required to be implemented by at least two Statistics Counters per Port.*	Count	**Optional DLLP Filters:** DLLP Type
5	**Any TLP Blocked by VL** –Number of Symbol times a TLP is blocked from transmission due to lack of VL flow control credits.	Count	**Optional Credit Filters:** TLP Type **Required Credit Filters:** VL
6	**Any TLP Blocked by (VH VL)** – Number of Symbol times a TLP is blocked from transmission due to the lack of (VH VL) flow control credits.	Count	**Optional Credit Filters:** TLP Type **Required Credit Filters:** VH, VL
7	**Any TLP Blocked by VL or (VH VL)** – Number of Symbol times a TLP is blocked from transmission due either to the lack of VL flow control credits or the lack of (VH VL) flow control credits.	Count	**Optional Credit Filters:** TLP Type **Required Credit Filters:** VH, VL

S Bit Number	Description	Count/Sample	Applicable Filter(s)
9	**All TLP Blocked by VL** –Number of Symbol times all TLPs are blocked from transmission due to lack of VL flow control credits (i.e., the lack of VL flow control credits results in no TLP being transmitted on the wire).	Count	**Optional Credit Filters:** TLP Type **Required Credit Filters:** VL
10	**All TLP Blocked by (VH VL)** – Number of Symbol times all TLPs are blocked from transmission due to the lack of (VH VL) flow control credits (i.e., the lack of (VH VL) flow control credits results in no TLP being transmitted on the wire).	Count	**Optional Credit Filters:** TLP Type **Required Credit Filters:** VH, VL
11	**All TLP Blocked by VL or (VH VL)** – Number of Symbol times all TLPs are blocked from transmission due either to the lack of VL flow control credits or the lack of (VH VL) flow control credits (i.e., the lack of VL or (VH VL) flow control credits results in no TLP being transmitted on the wire).	Count	**Optional Credit Filters:** TLP Type **Required Credit Filters:** VH, VL
32	**Available VL Transmit Credits** – Number of available transmit credits associated with a VL computed as: (CREDIT_LIMIT – CREDITS_CONSUMED) mod $2^{[Field Size]}$	Sample	**Required Credit Filters:** VL, Credit Type
33	**Available (VH VL) Transmit Credits** – Number of available transmit credits associated with a (VH VL) computed as: (CREDIT_LIMIT – CREDITS_CONSUMED) mod $2^{[Field Size]}$	Sample	**Required Credit Filters:** VL, VH, Credit Type
64	**Received TLPs** – Counts non-nullified TLPs received by this Port. *Required to be implemented by at least two Statistics Counters per Port.*	Count	**Optional TLP Filters:** VH, VL, TLP Type
65	**Received TLP DWORDs** – Counts DWORDs of non-nullified TLPs received by this Port. This includes framing symbols and all bytes received (i.e., STP to END inclusive). *Required to be implemented by at least two Statistics Counters per Port.*	Count	**Optional TLP Filters:** VH, VL, TLP Type
66	**Received Idle Symbols** – Counts the number of IDLE symbols received by this Port. *Required to be implemented by at least two Statistics Counters per Port.*	Count	None

S Bit Number	Description	Count/Sample	Applicable Filter(s)
67	**Received DLLPs** – Counts the number of DLLPs received by this Port. *Required to be implemented by at least two Statistics Counters per Port.*	Count	**Optional DLLP Filters:** DLLP Type
96	**Available VL Receive Credits** – Number of available receive credits associated with a VL computed as: (CREDITS_ALLOCATED – CREDITS_RECEIVED) mod $2^{[Field Size]}$	Sample	**Required Credit Filters:** VL, Credit Type
97	**Available (VH VL) Receive Credits** – Number of available receive credits associated with a (VH VL) computed as: (CREDITS_ALLOCATED – CREDITS_RECEIVED) mod $2^{[Field Size]}$	Sample	**Required Credit Filters:** VL, VH, Credit Type

4.5.2.2. Standard Filters

TLP Filtering consists of three filters: TLP Type, VL, and VH.

Table 4-73: TLP Filters

Bits	Filter	Description
0	TLP Type	Completion – If Set, Completion TLPs are not included (i.e., filtered out). If TLP Type filtering is not supported, this field is hardwired to zero.
1	TLP Type	Non-Posted – If Set, Non-Posted TLPs are not included (i.e., filtered out). If TLP Type filtering is not supported, this field is hardwired to zero.
2	TLP Type	Posted – If Set, Posted TLPs are not included (i.e., filtered out). If TLP Type filtering is not supported, this field is hardwired to zero.
15:3		Reserved
23:16	VH	VH Value – If VH filtering is enabled, this field contains the VH to include (i.e., filtered in). If VH filtering is not enabled, this field is ignored and all VHs are included. If VH filtering is not supported, this field is hardwired to zero.
26:24	VL	VL Value – IF VL filtering is enabled, this field contains the VL to include (i.e., filtered in). If VL filtering is not enabled, this field is ignored and all VLs are included. If VL filtering is not supported, this field is hardwired to zero.
29:27		Reserved
30	VH	VH Filter Enable – If Set, VH filtering is enabled. If Cleared, VH filtering is not enabled and all VHs are included. If VH filtering is not supported, this field is hardwired to zero.
31	VL	VL Filter Enable – If Set, VL filtering is enabled. If Cleared, VL filtering is not enabled and all VLs are included. If VL filtering is not supported, this field is hardwired to zero.

Credit Filtering consists of three filters: Credit Type, VL, and VH. Unsupported credit filters are hardwired to zero.

Table 4-74: Credit Filters

Bits	Filter	Description
0	Credit Type	Completion Header – If Set, Completion header Credits are not included (i.e., filtered out).
1	Credit Type	Non-Posted Header – If Set, Non-Posted header Credits are not included (i.e., filtered out).
2	Credit Type	Posted Header – If Set, Posted header Credits are not included (i.e., filtered out).
3	Credit Type	Completion Data – If Set, Completion Data Credits are not included (i.e., filtered out).
4	Credit Type	Non-Posted Data – If Set, Non-Posted Data Credits are not included (i.e., filtered out).
5	Credit Type	Posted Data – If Set, Posted Data Credits are not included (i.e., filtered out).
15:6		Reserved
23:16	VH	VH Value – Contains the VH to include (i.e. filtered in).
26:24	VL	VL Value – Contains the VL to include (i.e., filtered in).
31:27		Reserved

DLLP Filtering is optional and consists of a number of filters. Unsupported DLLP filters are hardwired to zero.

Table 4-75: DLLP Filters

Bits	Filter	Description
0	DLLP Type	Ack – If Set, Ack DLLPs are not included (i.e., filtered out).
1	DLLP Type	Nak – If Set, Nak DLLPs are not included (i.e., filtered out).
2	DLLP Type	Reset – If Set, Reset DLLPs are not included (i.e., filtered out).
3	DLLP Type	MRInit – If Set, MRInit DLLPs are not included (i.e., filtered out).
4	DLLP Type	Flow Control Initialization – If Set, the following DLLPs are not included (i.e., filtered out): InitFC1 InitFC2 MRInitFC1_VL MRInitFC1_VH MRInitFC2_VL MRInitFC2_VH
5	DLLP Type	Flow Control Update – If Set, UpdateFC and MRUpdateFC DLLPs are not included (i.e., filtered out).
6	DLLP Type	ASPM L1 – If Set, PM_Active_State_Request_L1 and PM_Request_Ack DLLPs are not included (i.e., filtered out).
7	DLLP Type	PM L1 L23 – If Set, PM_Enter_L1 and PM_Enter_L23 DLLPs are not included (i.e., filtered out).
8	DLLP Type	Vendor Specific – If Set, vendor Specific DLLPs are not included (i.e., filtered out).
31:9		Reserved

4.5.3. Statistics Block Table

This table contains an entry for each supported Statistics Block. Up to 32 Statistics Blocks may be supported by a component.

4.5.3.1. Statistics Block Capability (00h)

These fields describe the capabilities of the associated Statistics Block.

Table 4-76: Statistics Block Capability

Bit Location	Register Description	Attributes
1:0	**Statistics Block Status** – Indicates whether the statistics block is busy and, if so, whether the block is waiting or counting. Values in this field correspond to the states of the statistics collection process. **Values are:** 00b Idle 10b Waiting 11b Counting 01b Reserved Note that the upper bit of this field matches the value read from the Statistics Block Start/Busy register.	RO
15:2	**Reserved**	RO
31:16	**Statistics Table Size** – Indicates the number of entries contained in this Statistics Table associated with this Statistics Block.	RO

4.5.3.2. Statistics Table Offset (04h)

Table 4-77: Statistics Table Offset

Bit Location	Register Description	Attributes
3:0	**Reserved**	RO
31:4	**Statistics Table Offset** – Used as an offset from the address contained by one of the Function's Base Address registers to point to the base of the Statistics Table. The Base Address register used is selected by the Statistics Block BIR located in the MR-IOV Capability. This field is in units of 16-bytes (i.e. the whole register including the reserved bits is a byte offset to a structure that is 16-byte aligned).	RO

4.5.3.3. Statistics Wait Time (08h)

Table 4-78: Statistics Wait Time

Bit Location	Register Description	Attributes
15:0	**Waiting Period** – Indicates the time, in microseconds, of the waiting period. The waiting period is defined as the time from statistics collection initiation to the start of the counting period. A value of zero indicates no waiting period.	RW
31:16	**Reserved**	RO

4.5.3.4. Statistics Count Time (0Ch)

Table 4-79: Statistics Count Time

Bit Location	Register Description	Attributes
23:0	**Counting Period** – Indicates the time, in microseconds, of the counting period. The counting period is defined as the time during which selected events are counted and is the time from the end of the waiting period to the idle period. Note: A value of all ones (i.e., FFFFFFh) is interpreted as infinite. An infinite counting period ends when requested by software (i.e., the corresponding Statistics Block Start bit is Cleared). A value of zero corresponds to no counting period (this is useful for sampled values).	RW
31:24	**Reserved**	RO

4.5.4. Statistics Counter Table

Associated with each Statistics Block Table Entry is a Statistics Counter Table. The Statistics Counter Table contains one entry for each implemented Statistics Counter associated with a Statistics Block. The registers described in this section form a Statistics Counter Table entry.

4.5.4.1. Statistics Capability and Control (00h)

Table 4-80: Statistics Capability and Control

Bit Location	Register Description	Attributes
7:0	**Port Number** – Indicates which Port the counter is associated with. For a Device, this field must be zero. For a Switch, this value contains an index into the Port Table.	RO
15:8	**Statistics Descriptor Index** – Contains the index of the Statistics Descriptor Table entry that describes the statistics recording capabilities of this counter.	RO
21:16	**Counter Width** – Indicates the width of the counter. Value is counter width-1 (i.e., a 32-bit counter contains 31 in this field). If the counter supports any standard statistics, the implemented counter width must be 32 bits or greater. If the counter supports only Vendor Specific statistics, the counter width can be any value.	RO
22	**Reserved**	RO
23	**Counter Enable** – When Set, the counter is enabled. When Cleared, the counter is disabled and certain fields (defined below) are undefined. Software should disable unused counters to reduce power consumption. This bit is allowed to be hardwired to 1b if the counter is always enabled. The default value of this field is Vendor Specific.	RW
31:24	**Statistics Select** –Determines what statistic software has selected for this counter to record. This field contains the bit number of one of the supported bits of the Statistics Descriptor entry selected by Statistics Descriptor Index. The counter value is undefined if the value of this field indicates an unsupported counter (i.e., the value does not correspond to an S bit or the associated S bit is 0b). Bits in this field may be Read Only if they are not needed. For example, a Statistics Descriptor Index that supports S bits 16, 17, and 18 need only implement this field as 000100WWb where W represents read/write bits. Following this rule to its extreme, if a Statistics Descriptor Index supports exactly one S bit, this entire field may be Read Only. The default value of this field is Vendor Specific. This field is undefined when the counter is disabled.	RW

4.5.4.2. Statistics Filter Enable and Control (04h)

Table 4-81: Statistics Filter Enable and Control

Bit Location	Register Description	Attributes
31:0	**Filter Enable and Control** – The meaning of this field depends on the Statistics Style and Statistics Select fields. The description of each counter defines the meaning of the filters associated with it. This field is undefined when the counter is disabled. The default value of this field is Vendor Specific.	RW

4.5.4.3. Statistics Counter Low (08h)

Table 4-82: Statistics Counter Low

Bit Location	Register Description	Attributes
Width:0	**Count Value Low** – Indicates the lower 32 bits of the counter value. Unused bits, as indicated by Counter Width, are Read Only Zero. The counter value is undefined when the counter is disabled or Busy (i.e., in the waiting or counting period). The default value of this field is Vendor Specific.	RO
31:Width	**Reserved**	RO

4.5.4.4. Statistics Counter High (0Ch)

Table 4-83: Statistics Counter High

Bit Location	Register Description	Attributes
Width-32:0	**Count Value High** – Indicates the upper bits of the counter value. Unused bits, as indicated by Counter Width, are Read Only Zero. The counter value is undefined when the counter is disabled or Busy (i.e., in the waiting or counting period). The default value of this field is Vendor Specific.	RO
31:Width-32+1	**Reserved**	RO

5

Error Handling

5. PCIe Error Mapping to MR

The basic rules for error detection, logging, and reporting are unchanged from PCIe. The only change is which VH(s) should be affected.

Table 5-1: Physical Layer Error List

Error Name	Error Type	Detecting Agent Action (PCIe)	Detecting Agent Action (MR IOV)
Receiver Error	Correctable	*Receiver (if checking):* Send ERR_COR to Root Complex.	*Receiver:* Same as PCIe but send on all enabled VHs not in Reset.

Table 5-2: Data Link Layer Error List

Error Name	Error Type	Detecting Agent Action (PCIe)	Detecting Agent Action (MR IOV)
Bad TLP	Correctable	*Receiver:* Send ERR_COR to Root Complex.	*Receiver:* Same as PCIe but send to all enabled VHs not in Reset.
Bad DLLP	Correctable	*Receiver:* Send ERR_COR to Root Complex.	*Receiver:* Same as PCIe but send on all enabled VHs not in Reset.
Replay Timeout	Correctable	*Transmitter:* Send ERR_COR to Root Complex.	*Transmitter:* Same as PCIe but send on all enabled VHs not in Reset.
REPLAY NUM Rollover	Correctable	*Transmitter:* Send ERR_COR to Root Complex.	*Transmitter:* Same as PCIe but send on all enabled VHs not in Reset.
Data Link Layer Protocol Error	Uncorrectable (Fatal)	If checking, send ERR_FATAL to Root Complex.	Same as PCIe but send to all enabled VHs not in Reset.

Table 5-3: Transaction Layer Error List

Error Name	Error Type	Detecting Agent Action (PCIe)	Detecting Agent Action (MV IOV)
Poisoned TLP Received	Uncorrectable (Non-Fatal)	*Receiver:* Send ERR_NONFATAL to Root Complex, or ERR_COR for the Advisory Non-Fatal Error cases. Log the header of the Poisoned TLP.	*Receiver:* Same as PCIe Error message sent only for affected VH. Log the header for the affected VH only.
ECRC Check Failed	Uncorrectable (Non-Fatal)	*Receiver (if ECRC checking supported):* Send ERR_NONFATAL to Root Complex, or ERR_COR for the Advisory Non-Fatal Error case. Log the header of the TLP that encounter the ECRC error.	*Receiver (if ECRC checking supported):* Same as PCIe Error message sent only for affected VH. Log the header for the affected VH only.
Unsupported Request (UR)	Uncorrectable (Non-Fatal)	*Request Receiver:* Send ERR_NONFATAL to Root Complex, or ERR_COR for the Advisory Non-Fatal Error case. Log the header of the TLP that caused the error.	*Request Receiver:* Same as PCIe Error message sent only for affected VH. Log the header for the affected VH only.
Completion Timeout	Uncorrectable (Non-Fatal)	*Requester:* Send ERR_NONFATAL to Root Complex, or ERR_COR for the Advisory Non-Fatal Error case .	*Requester:* Same as PCIe Error message sent only for affected VH.
Completer Abort	Uncorrectable (Non-Fatal)	*Completer:* Send ERR_NONFATAL to Root Complex, or ERR_COR for the Advisory Non-Fatal Error case. Log the header of the Request that encountered the error.	*Completer:* Same as PCIe Error message sent only for affected VH. Log the header for the affected VH only.

Error Name	Error Type	Detecting Agent Action (PCIe)	Detecting Agent Action (MV IOV)
Unexpected Completion	Uncorrectable (Non-Fatal)	*Receiver:* Send ERR_COR to Root Complex. This is an Advisory Non-Fatal Error. Log the header of the Completion that encountered the error.	*Receiver:* Same as PCIe Error message sent only for affected VH. Log the header for the affected VH only.
Receiver Overflow	Uncorrectable (Fatal)	*Receiver (if checking):* Send ERR_FATAL to Root Complex.	*Receiver (if checking):* Same as PCIe Send ERR_FATAL to all Root Complex that have a VC on the affected Link mapped to the affected VL.
Flow Control Protocol Error	Uncorrectable (Fatal)	*Receiver (if checking):* Send ERR_FATAL to Root Complex.	*Receiver (if checking):* Same as PCIe Send ERR_FATAL to all Root Complex that have a VC on the affected Link mapped to the affected VL.
Malformed TLP	Uncorrectable (Fatal)	*Receiver:* Send ERR_FATAL to Root Complex. Log the header of the TLP that encountered the error.	*Receiver:* Same as PCIe Error message sent only for affected VH. Log the header for the affected VH only.

5.1. MR Errors

Errors associated with MR-IOV operation are listed in Table 5-4.

When a single TLP triggers more than one error, a single error corresponding to the highest priority error is raised. Priority is as follows:

Highest — MR Uncorrectable Fatal Error

MR Uncorrectable Non-Fatal Error

MR Uncorrectable Global Key Error

Lowest — MR Correctable Global Key Error

Table 5-4: MR Error List

Error Name	Error Type	Detecting Agent Action (PCIe)	Detecting Agent Action (MR IOV)
TLP received without MR Prefix	Uncorrectable Fatal	N/A	*Receiver:* Signals MR Uncorrectable Fatal TLP Error, discard TLP, so not update flow control (VL/VH cannot be trusted).
Invalid TLP Prefix	Uncorrectable Fatal	N/A	*Receiver:* Signal MR Uncorrectable Fatal TLP Error, discard TLP, do not update flow control (VL/VH cannot be trusted).
TLP received on {Port, VH} where VH > Port's NumVH	Uncorrectable Fatal	N/A	*Receiver:* Signal MR Uncorrectable Fatal TLP Error, discard TLP, do not update flow control (VL/VH cannot be trusted).
TLP received on VH in reset (sender of TLP has Acked entering Reset)	Uncorrectable Non-Fatal	N/A	*Receiver:* Signal MR Uncorrectable Non-Fatal TLP Error, discard TLP, update flow control normally.
TLP received on {Port, VH} that is not mapped to any VS Bridge	Uncorrectable Non-Fatal	N/A	*Receiver:* Signal MR Uncorrectable Non-Fatal TLP Error, discard TLP, update flow control normally.
TLP Prefix with Global Key Mismatch: TLP at destination or forwarded on PCIe Link	Uncorrectable Global Key	N/A	*Receiver:* Signal MR Uncorrectable Global Key Error, discard TLP, update flow control normally.
TLP Prefix with Global Key Mismatch: TLP being forwarded on MR Link	Correctable Global Key	N/A	*Receiver:* Signal MR Correctable Global Key Error, forward TLP normally.

Error Name	Error Type	Detecting Agent Action (PCIe)	Detecting Agent Action (MR IOV)
TLP Prefix (VH VL) that is invalid, not enabled or has not finished Flow Control Initialization	Uncorrectable Fatal	N/A	*Receiver:* Signal MR Uncorrectable Fatal TLP Error, discard TLP, do not update flow control (VL/VH cannot be trusted).
MRUpdateFC for (VH VL) that is invalid	Correctable	N/A	*Receiver:* Signal MR DLLP Error, discard DLLP.
MRUpdateFC for (VH VL) where the VH is not in reset and either the (VH VL) is not enabled or is enabled but has not completed Flow Control Initialization.	Correctable	N/A	*Receiver:* Signal MR DLLP Error, discard DLLP.
Invalid VH Group in Reset DLLP	Correctable	N/A	*Receiver:* Signal MR DLLP Error, discard DLLP.
Out of range Assert bit set in Reset DLLP	Correctable	N/A	*Receiver:* Signal MR DLLP Error, ignore the offending Assert bit(s) and process remainder of Reset DLLP normally.

6

6. Hot Plug

6.1. MRA Switch

The Virtual Signals Interface registers in the VS Bridge Table provide the MR-PCIM end of the "Virtual Hot-Plug Signals Interface." This section describes the connection between those fields and the PCI Express Slot registers of the corresponding Type 1 header.

If the Bridge Controls Physical Link field in the VS Bridge Table is Clear, the VS Type 1 header, slot registers are virtual and connect to the "Virtual Hot-Plug Signals Interface" described in Section 4.3.6.3.

If the Bridge Controls Physical Link field in the VS Bridge Table is Set, the VS Type 1 header fields are physical and the external signals of the Switch (if implemented). Specifically, in this mode, the Type 1 header slot registers and the slot registers in the Port Table are identical. Changes to one register affects the other, external events are visible in both, there is a single set of "changed" bits and write 1 to clear to either register clears then.

The Type 1 header associated with each virtual downstream Switch Port shall contain the PCIe Slot Capabilities, Control, and Status registers. These registers work exactly as in base PCIe as described in Sections 7.8.9, 7.8.10, and 7.8.11 of the *PCI Express Base Specification*.

For each Type 1 header register, the following sections describe how the various bits interact with the "Virtual Hot-Plug Signals Interface." Figures are extracted for reference from the *PCI Express Base Specification*.

Changes to the Virtual Hot-Plug Controller occur immediately. If Bridge Controls Physical Link is Clear, the No Command Complete Support bit in the Type 1 header is always Set.

6.1.1. PCI Express Capability: Slot Capability Register

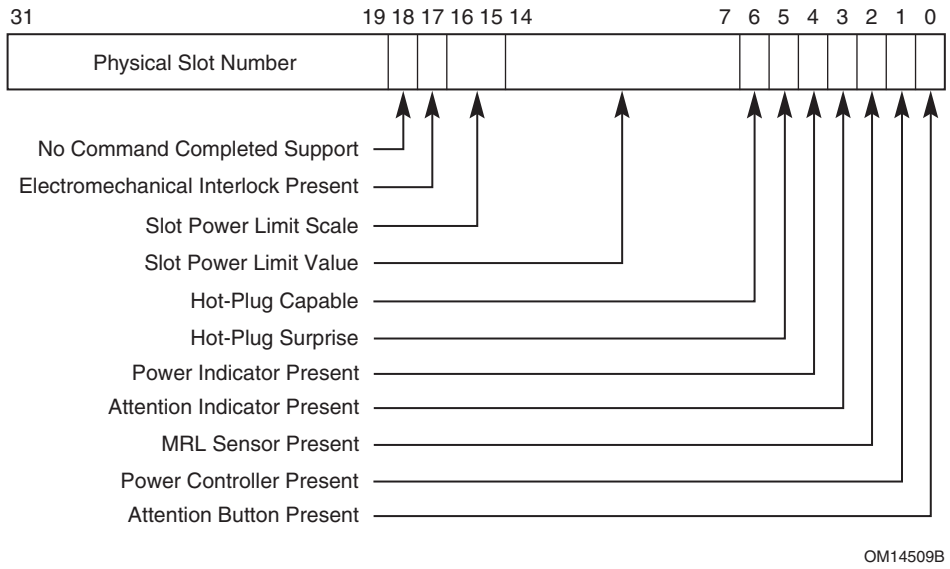

Figure 6-1: Slot Capabilities Register

Table 6-1: Virtual Mapping: PCIe Slot Capabilities Register

Bits	Field Name	If Virtual (Bridge Controls Physical Link = 0b)	If Physical (Bridge Controls Physical Link = 1b)
0	Attention Button Present		Base (HWInit)
1	Power Controller Present	Bit 1 Signals Interface 1, Power Controller Present	Base (HWInit)
2	MRL Sensor Present	0b	Base (HWInit)
3	Attention Indicator Present	1b	Base (HWInit)
4	Power Indicator Present	1b	Base (HWInit)
5	Hot-Plug Surprise	Bit 5 Signals Interface 1, Virtual Hot Plug Surprise	Base (HWInit)
6	Hot Plug Capable	Bit 6 Signals Interface 1, Virtual Hot Plug Capable	Base (HWInit)
14:7	Slot Power Limit Value	Bits 14:7 Signals Interface 1, Virtual Slot Power Limit Value	Base (HWInit)
16:15	Slot Power Limit Scale	Bits 16:15 Signals Interface 1, Virtual Slot Power Limit Scale	Base (HWInit)

Bits	Field Name	If Virtual (Bridge Controls Physical Link = 0b)	If Physical (Bridge Controls Physical Link = 1b)
17	Electromechanical Interlock Present	0b	Base (HWInit)
18	No Command Completed Support	0b	Base (HWInit)
31:19	Physical Slot Number	Bits 31:19 Signals Interface 1, Virtual Slot Number	Base (HWInit)

6.1.2. PCI Express Capability: Slot Control Register

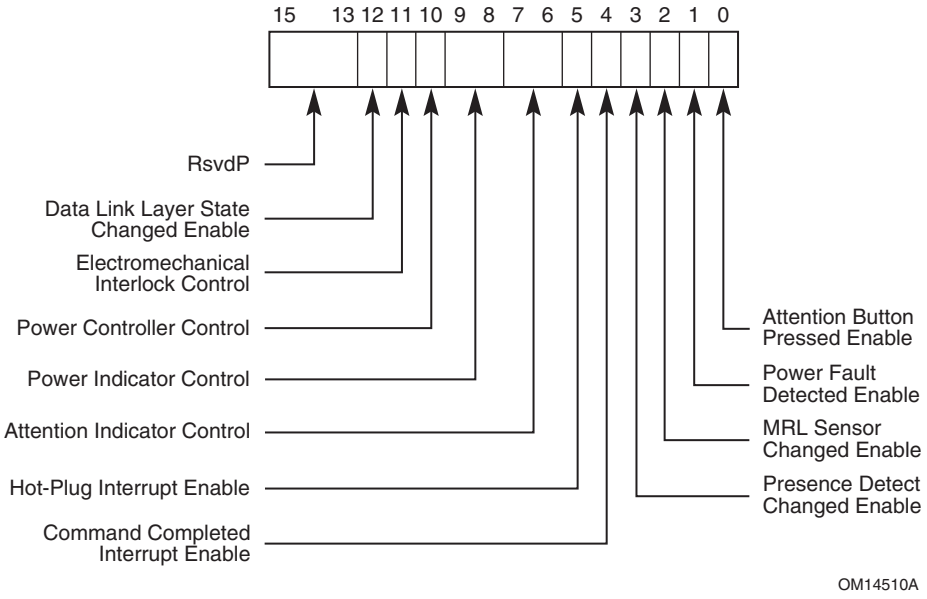

Figure 6-2: Slot Control Register

Table 6-2: Virtual Mapping: PCIe Slot Control Register

Bits	Field Name	If Virtual (Bridge Controls Physical Link = 0b)	If Physical (Bridge Controls Physical Link = 1b)
0	Attention Button Pressed Enable	Implement	Base
1	Power Fault Detected Enable	Implement	Base
2	MRL Sensor Changed Enable	0b	Base
3	Presence Detect Changed Enable	Implement	Base
4	Command Completed Interrupt Enable	0b	Base
5	Hot-Plug Interrupt Enable Note: This bit controls Interrupts delivered in the VH. It is unrelated to the similarly named Hot-Plug Signals Interrupt Enable bit. The latter governs interrupts delivered to MR-PCIM in the Management VS.	Implement	Base
7:6	Attention Indicator Control	Bits 3:2, Signals Interface 2, Virtual Attention Indicator State	Base
9:8	Power Indicator Control	Bits 1:0, Signals Interface 2, Virtual Power Indicator State	Base
10	Power Controller Control	Bit 4 Signals Interface 2, Virtual Power Controller State Turning off Virtual Power State also causes the associated Link to see a VH Reset (this is the same effect as setting Secondary Bus Reset).	Base
11	Electromechanical Interlock Control	0b	Base
12	Data Link Layer Changed Enable	Implement	Base

Bits	Field Name	If Virtual (Bridge Controls Physical Link = 0b)	If Physical (Bridge Controls Physical Link = 1b)
15:13	Reserved	0b	Base

6.1.3. PCI Express Capability: Slot Status Register

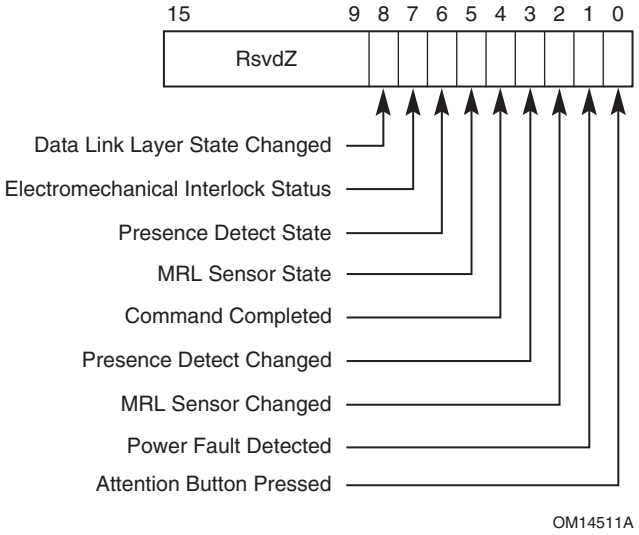

Figure 6-3: Slot Status Register

Table 6-3: Virtual Mapping: PCIe Slot Status Register

Bits	Field Name	If Virtual (Bridge Controls Physical Link = 0b)	If Physical (Bridge Controls Physical Link = 1b)
0	Attention Button Pressed	Set if 1b written to Bit 24, Signals Interface 2, Push Virtual Attention Button	Base
1	Power Fault Detected	Set if 1b written to Bit 25, Signals Interface 2, Signal Virtual Power Fault	Base
2	MRL Sensor Changed	0b	Base
3	Presence Detect Changed	Set on transition of Bit 17, Signals Interface 2, Virtual Presence Detect State	Base
4	Command Completed	0b	Base
5	MRL Sensor State	0b	Base

Bits	Field Name	If Virtual (Bridge Controls Physical Link = 0b)	If Physical (Bridge Controls Physical Link = 1b)
6	Presence Detect State	Bit 17, Signals Interface 2, Virtual Presence Detect State	Base
7	Electromechanical Interlock Status	0b	Base
8	Data Link Layer State Changed	Set on transition of Bit 16, Signals Interface 2, Virtual Data Link State	Base
15:9	Reserved	0b	Base

6.1.4. PCI Express Capability: Device Capabilities Register

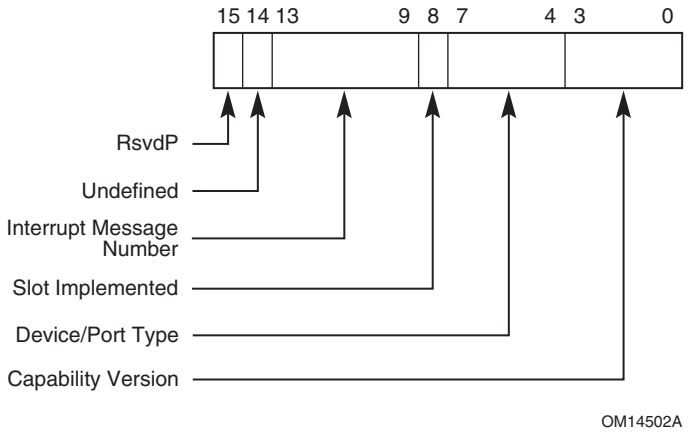

Figure 6-4: PCI Express Capabilities Register

Table 6-4: Virtual Mapping: PCIe Capabilities Register

Bits	Field Name	If Virtual (Bridge Controls Physical Link = 0b)	If Physical (Bridge Controls Physical Link = 1b)
3:0	Capability Version	Base	Base
7:4	Device/Port Type	Base	Base
8	Slot Implemented	Bit 8, Signals Interface 2, Virtual Slot Implemented	Base
13:9	Interrupt Message Number	Base	Base
14	Undefined	Base	Base

Bits	Field Name	If Virtual (Bridge Controls Physical Link = 0b)	If Physical (Bridge Controls Physical Link = 1b)
15	Reserved	Base	Base

6.1.5. Hot-Plug Virtual Signals Interface Registers

Each virtual downstream Switch Port has a Virtual Signals Interface as defined in Section 4.3.6.3. Registers controlling this hardware are located in the VS Bridge Table. They provide the interface between MR-PCIM Hot-Plug software and Hot-Plug software running in the VH. This interface allows MR-PCIM to:

1. Provide configuration and status information to the virtual slot registers.
2. Push virtual buttons of the Virtual Hot-Plug controller (i.e., change bits in the Virtual Slot Status register).
3. Detect Indicator and Control changes made to the Virtual Hot-Plug controller (i.e., detect certain changes to the Virtual Slot Control and Virtual Slot Capabilities registers).

Table 6-5 lists the register fields that are defined for this function.

Table 6-5: Hot-Plug Virtual Signals Interface Register Fields

Bit Field	MR-PCIM View	Purpose
Virtual Slot Number	Read/Write	1
Virtual Slot Power Limit Scale/Value	Read Only	3
Virtual Slot Implemented	Read/Write	1
Virtual Hot-Plug Capable	Read/Write	1
Virtual Hot-Plug Surprise	Read/Write	1
Virtual Data Link State	Read/Write	2
Virtual Power Controller State	Read Only	3
Virtual Power Controller State Changed	Read/Write One to Clear	3
Virtual Power Controller Present	Read/Write	1
Virtual Power Indicator State	Read Only	3
Virtual Power Indicator State Changed	Read/Write One to Clear	3
Virtual Attention Indicator State	Read Only	3
Virtual Attention Indicator State Changed	Read/Write One to Clear	3
Virtual Presence Detect State	Read/Write	2
Press Virtual Attention Button	Read Zero, Write One to Set	2
Signal Virtual Power Fault	Read Zero, Write One to Set	2

These fields are more precisely defined in Section 4.3.6.3.

In addition to the above fields, changes by VH software to the Attention Indicator, Power Indicator, and Power Controller Control (Purpose 3 above) can generate an interrupt to MR-PCIM.

6.1.6. Physical Slot Registers

Each physical slot has an associated set of registers for controlling the Physical Hot Plug Controller. As in PCIe, the Physical Hot-Plug Controller is optional.

If present, the Physical Hot Plug Controller is managed via the Slot registers of the associated Port Table entry.

In addition, when a Bridge Controls Physical Link field in the VS Bridge Table is Set, the Physical Hot-Plug Controller for the Port mapped to that VS Bridge Table entry is also controlled using the Slot Capability, Control, and Status registers of the associated with the Type 1 header. This is useful when an MRA Switch is being used as a Base PCIe Switch. It is also useful if MR-PCIM chooses to delegate authority for managing the Physical Link to software running in a specific VH (typically because the associated Port is attached to a Base PCIe Device).

6.1.7. Physical Hot-Plug Signals Interface

Hot-Plug support is optional in PCI Express. If provided, some means for communicating Hot-Plug signals from the Switch is needed. In PCI Express, this mechanism is Vendor Specific.

Physical Hot-Plug remains optional in MRA Switches. The Physical Signals Interface also remains Vendor Specific.

The presence or absence of Hot-Plug support is indicated using the Slot Implemented bit in the Port Table. If present, various Hot-Plug signals are optional and their presence is indicated in the Slot Capabilities register also in the Port Table.

If the Bridge Controls Physical Link field in some VS Bridge Table entry is Set, the Physical Hot-Plug information for the Port mapped to that VS Bridge is also reflected in the associated Type 1 header.

6.2. Virtual Device Migration

Virtual Hot Plug can be used to support Device Migration from one VH to another VH.

To accomplish this, the losing VH gets a Hot Remove sequence. This sequence starts with a push of the Virtual Attention Button and ends with the disabling of the Virtual Power Controller.

Software could then remap and reset the Virtual Device by:

1. Ensuring that the gaining VS Bridge is ready to receive the Virtual Device (i.e., the Port/Port VHN fields are unmapped and the Data Link State is clear).
2. Clearing the Port/Port VHN fields in the losing VS Bridge Table entry.
3. Setting the Force Reset bit in the gaining VS Bridge Table entry.

4. Mapping the Port/Port VHN fields into the gaining VS Bridge Table Entry.

Software would then send the gaining VH a Hot Add sequence. This starts with the assertion of Presence Detect and finishes with the Device being enumerated and used by software in the VH.

6.3. Base PCI Express Device Migration

Base PCIe components can also be attached to a Switch and assigned to a single VH. This assignment can change over time through a Device Migration process.

The sequence of operations is similar to that described in Section 6.2. The exception is that since the Link is operating in PCIe mode, the Secondary Bus Reset field in the Port Table should be used instead of the Force Reset field from the VS Bridge Table to cleanse Device state for the gaining VH.

7

7. Power Management

MR systems continue to need power management capabilities. Two varieties of power management are involved:

- Virtual power management allows software running in a VH to believe that it has turned off power to one or more virtual functions.
- Actual power management allows MR-PCIM software to control device power.

7.1. Overview

ASPM and PCI-PM are expanded for MR-IOV. MR Components have both virtual and physical D-states. Slots have virtual and physical power states. Virtual and physical ASPM controls also exist.

7.2. Virtual D-State

Every Function in every VH of every MR Device or Switch has a virtual D-state. This includes BFs, PFs, VFs, and Functions as well as P2P Bridges.

Virtual D-state is controlled by software operating in each VH using the rules defined in the *PCI Bus Power Management Interface Specification, Revision 1.2*, and in the *PCI Express Base Specification.*

Component D-state is an extension of the PCIe multi-function component rules. An MR Component is treated as a multi-function component taking the Functions in all VHs into account. For example, a shared component may not enter L1 state until all Functions in all VHs of the component have been written to non-D0 states.

When software in a VH writes a Function to a lower power virtual D-state, the component acts as if it were in a lower power state. Affects on actual power consumption are vendor specific. Vendors are encouraged to use this mechanism to reduce power consumption whenever possible.

Because the component may have not powered down, lower virtual D-states may not result in actual power savings.

7.3. Link Power States

Link power states and transitions are unchanged from PCIe. L1 and L2/L3 handshakes are unchanged. As in PCIe, Link state is affected by the D-state of all Functions in the Component. In MR, this includes all Functions in all VHs. There are exceptions to deal with the D-state of Functions used for managing the MR topology and D-states of Functions in VHs that are not being used.

In the *PCI Express Base Specification*, a Switch Upstream Port may enter L0s when all of the Switch's Downstream Port Receive Lanes are in the L0s state and a Switch Downstream Port may enter L0s when the Switch's Upstream Port's Receive Lanes are in the L0s state.[15]

For MR-IOV, a single Port may be both Upstream and Downstream on different VHs and a single Port may have VHs that are in Reset as well as VHs that are not in Reset. An MR Switch Port may enter L0s only if all VS Bridges mapped to the Port may enter L0s. Any VS Bridge that is in Reset as indicated by the Link in Reset bit (see Section 4.3.6.1) is treated as being in the L0s state for this determination.

For example, a Port that is mapped as the Upstream Bridge of VS1 and VS3 and a Downstream Bridge in VS2, that is receiving Hot Reset on the VH mapped to the Upstream Bridge of VS3, may enter L0s when (1) the Receive Lanes for all Ports mapped as Downstream Bridges in VS1 are in the L0s state, (2) the Receive Lanes for the Port mapped as the Upstream Bridge of VS2 is in the L0s state, (3) all Downstream Bridges of VS3 have their Link in Reset bit Ser, (4) there are no TLPs pending for transmission over this Port or no VF credits are available to transmit any TLPs on this Port, and (5) no DLLPs are pending for transmission on this Port.[16]

For Switches, ASPM is controlled using the ASPM Control field in the Port Table. The ASPM Control field in the PCI Express Capability is ignored.

For Devices, ASPM is controlled using the ASPM Control field associated with all Functions in VH0. The ASPM Control field in the PCI Express Capability of non-zero VHs is ignored.

7.4. Multi-Root ASPM

For Switches, ASPM is controlled by MR-PCIM using the Port Table (see Section 4.3.3.12 for details). ASPM may also be controlled by the Type 1 header in each VH if the Bridge Controls Physical Link bit is Set in the associated VS Bridge Table entry (see Section 4.3.6.2).

When the Bridge Controls Physical Link bit is clear, virtual ASPM controls are provided for software compatibility but perform no function. Like PCIe, virtual ASPM controls must support L0s. L1 ASPM support is optional in PCIe and, for simplicity, virtual L1 ASPM is not supported.

For Devices, ASPM is controlled using the ASPM Control field associated with all Functions in VH0. The ASPM Control field in the PCI Express Capability of non-zero VHs is ignored..

7.5. Slot Clock and Common Clock Configuration

The Slot Clock Configuration and Common Clock Configuration bits in Type 1 headers associated with Base PCIe Ports reflect the physical Port.

---

[15] The PCI Express Base Specification contains additional conditions for L0s entry that are common between Upstream and Downstream Switch Ports.

[16] Conditions 4 and 5 are from the *PCI Express Base Specification*.

The Slot Clock Configuration and Common Clock Configuration bits in Type 1 headers associated with MR Ports reflect the virtual environment and always indicate a common clock. The underlying physical configuration is available in the Port table (see Section 4.3.3.12 for details).

7.6. Multi-Root Wake-Up

The wake-up model is mostly unchanged from PCIe. The exception is the need to deal with the "power inversion" situations. In PCIe, power is turned off starting at the leaves of a PCIe hierarchy so that power is removed from a root only after power was first removed from all components below that root. In MR topologies, shared components have virtual power removed while the non-shared components have actual power removed. This creates situations where a powered on shared component located below a non-shared component (in some VH), needs to wake-up that non-shared component. It also creates situations where a powered off component needs to wake-up a powered on shared component.

Examples of these scenarios are shown in Figure 7-1.

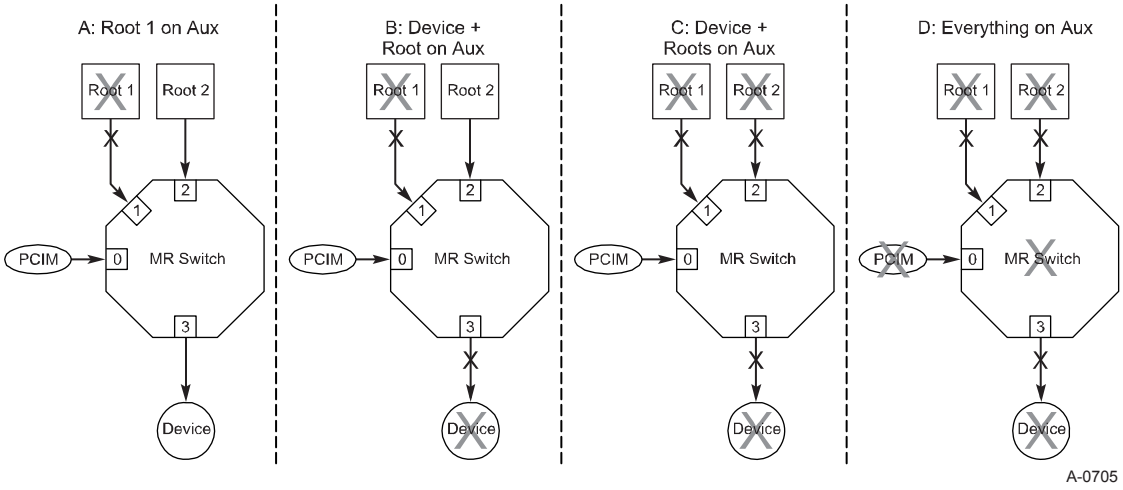

Figure 7-1: Multi-Root Wake-Up Scenarios

To address this, MR Switches implement a number of power management features:

- The ability to send Beacon/WAKE# on receipt of certain PM_PME messages.
- The ability to detect Beacon/Wake# and convert it into a PM_PME message.
- The ability to detect Beacon/WAKE# and signal an interrupt.

7.6.1. PME Triggers Beacon/Wake#

The PME Triggers Beacon/WAKE# bit in the Port Table instructs an MR Switch to issue Beacon or WAKE# when a PM_PME Message is headed out a Port that is DL_Down.

Scenario A: This mechanism can be used to wake-up Root 1 in Figure 7-1 Scenario A. The MR Device detects a wake-up event in one of its Functions. The Link from the Device to the MR

Switch is in DL_Active and the Device is being used by Root 2. The Device sends a PM_PME Message upstream in the VH associated with Root 1. The MR Switch Port 3 receives the PM_PME and because it is associated with Root 1, forwards it to Port 1.

Port 1 is DL_Down and has its PME Triggers Beacon/WAKE# bit Set. This causes the PM_PME Message to be queued and a wake-up event to be sent to Root 1 (as in PCIe, whether this involves Beacon or WAKE# is platform specific).

7.6.2. Beacon/Wake# Triggers MSI

The Beacon/WAKE# Interrupt Pending and Beacon/WAKE# Interrupt Enable bits in the Port Table instructs an MR Switch to generate an MSI interrupt to MR-PCIM when it detects a Beacon or WAKE# indication from the Port.

Scenario B: This mechanism is used to power up the Device in Figure 7-1 Scenario B. The Device detects a wake-up event in one of its Functions. The Device is operating on Aux power and generates a Beacon or WAKE# to the MR Switch. The MR Switch interrupts MR-PCIM which notices the Beacon/WAKE# event and powers up the Device using the Physical Power Controller associated with Port 3. Once the Link comes up, Scenario A applies.

Scenario C: This mechanism is also used to power up the Device in Figure 7-1 Scenario C. The Device is powered up as in Scenario B. When the PM_PME Message is sent, it is addressed to either Root 1 or Root 2 and Scenario A applies to the addressed Root.

7.6.3. Beacon/WAKE# Triggers Beacon/WAKE#

An MR Switch that is operating on Aux power will broadcast a Beacon or WAKE# indication received on a Downstream Port to all Authorized Upstream Ports.

Scenario D: This mechanism is used to power up everything in Figure 7-1 Scenario D. The Port containing MR-PCIM is Authorized. The Device is operating on Aux power and generates a Beacon or WAKE# to the MR Switch. The MR Switch generates Beacon or WAKE# out of the Authorized Port headed to MR-PCIM. The Root where MR-PCIM is powered on which, in turn, powers on the MR Switch. After the MR Switch is powered on and configured, Scenario C applies.

7.7. Multi-Root PME Turn Off

MR Devices respond to PME_Turn_Off messages with PME_TO_Ack within each VH. In VH0, PME_Turn_Off messages will cause a Downstream Component to request a Link transition to L2/L3 Ready using the PM_Enter_L23 DLLP. In non-zero VHs, PME_Turn_Off has no effect on Link state. Devices must cleanse state in all Functions of a VH before sending PME_TO_Ack. This ensures that virtual Device state disappears when the device is "powered off."

MR Switches process PME_Turn_Off messages using PCIe rules within each VS. When a PME_Turn_Off message is received at the Upstream Bridge of a VS, the message is broadcast to all Downstream Bridges of the VS. When the downstream sees the downstream component responding with PME_TO_Ack, the responses are recorded in a scoreboard. When the last Downstream Bridge responds, a PME_TO_Ack message is sent Upstream.

For Upstream Links where the Link's Bridge Controls Physical Link is 1b, MR Switches will request the Link to transition to L2/L3 Ready using the PM_Enter_L23 DLLP following completion of the PME_Turn_Off/PME_TO_Ack handshake.

For Upstream Links where the Link's Bridge Controls Physical Link is 0b, MR Switches will not automatically transition the Link to the L2/L3 Ready state. Instead the entry into L2/L3 Ready state is controlled using the Send PM_Enter_L23 DLLP bit in the Port Table. See Section 4.3.3.2 for details.

The PME Turn Off State Change Interrupt feature in the VS Bridge Table can be used to determine which VHs on a Port have completed the PME Turn Off handshake (see Sections 4.3.6.1 and 4.3.6.2). When the appropriate Bridges have completed their handshake, software can Set the Send PM_Enter_L23 DLLP bit for the affected Upstream Link.

7.8. Multi-Root Power Controller

MR-PCIM may enable a virtual power controller for each virtual slot by setting the Virtual Power Controller Present bit in the Hot-Plug Virtual Signals Interface (see Section 4.3.6.3). If this bit is Set and the Bridge Controls Physical Link bit is Clear, when the VH turns off power using the PCIe Hot-Plug controller, the VH is sent a reset.

The virtual power controller has no effect on physical power. Turning off virtual power causes the MR Switch to send Reset DLLPs to cleanse state from the affected Components.

A form factor can allow MR-PCIM to control the physical power to a slot. As in PCIe Base, doing so is optional. This control occurs through the Port Table (see Section 4.3.3.12 for details).

If the Bridge Controls Physical Link bit is Set, the virtual power controller in the VS Bridge becomes the physical power controller and directly controls power as defined in the *PCI Express Base Specification*.

7.9. Multi-Root Power Budgeting

Power Budgeting is optional in the *PCI Express Base Specification*. Certain form factors may require it. Power Budgeting is required for Devices supporting Hot-Plug. See Section 7.15 of the *PCI Express Base Specification*.

Power Budgeting remains optional in MR under the same conditions. If provided, power values reflect the power consumed by the BF and all associated PFs and VFs in all VHs.

8

8. Congestion Management

Exceeding the bandwidth of a Link or capacity of a buffer can lead to congestion in a topology. In a Multi-Root Topology, congestion may affect the performance of unrelated VHs and lead to Completion Timeout errors. This chapter defines mechanisms for detecting and controlling congestion in a Multi-Root Topology.

8.1. Overview

There are three possible causes of congestion in a PCIe topology:

- A fault in hardware or software configuration of a device in the topology.
- A static rate mismatch in the capacity of the path from a component injecting traffic into the topology (e.g., a Device) and the ultimate destination (e.g., a Root Port). Congestion due to a static rate mismatch would occur even if the topology were otherwise idle.
- Traffic merging of multiple flows, none of which individually suffer from a static rate mismatch, causing the capacity of an element in the topology to be exceeded.

While the causes of congestion are the same in both single and multi-root PCIe topologies, it is desirable to provide the ability to manage, limit, and contain the congestion caused by one VH on other VHs in the system. The congestion management mechanisms outlined in this section allow management of congestion that is unique to MR topologies (i.e., due to traffic merging from different VHs). These mechanisms do not address congestion within a VH since this congestion would have been present in an equivalent PCIe Base topology.

MR congestion management mechanisms provide the following benefits.

- Preserve the behavior of Virtual Channels (VCs) defined by the *PCI Express Base Specification* within a VH.
- Allow systems to be constructed where a fault in one VH does not result in errors (e.g., Completion Timeouts) in another VH.
- Allow systems to be constructed that support forward progress guarantees on a VH or groups of VHs when congestion exists on an unrelated VH.
- Support a wide range of implementation options. At one extreme, they allow creation of MR Devices through incremental changes to SR Devices. At the other extreme, they allow implementations that support complete isolation between virtual hierarchies.

8.2. Congestion Isolation

The Virtual Link (VL) mechanism together with Bypass Queues, a logical queuing structure associated with a VL at a Receiver, provide the foundation for isolating congestion and supporting differentiated services within a Multi-Root PCI Express Topology. This section defines these mechanisms and describes them from a system perspective.

8.2.1. Virtual Links

The Virtual Link (VL) mechanism provides the means to support multiple independent logical data flows over a single physical Multi-Root PCIe Link. VLs play the same role in an MR topology as Virtual Channels (VCs) in a PCIe Base topology. VLs are associated with independent fabric resources (queues/buffers and associated control logic) that are used to move information across an MR Link with independent flow control. As with VC in a PCIe Base topology, VLs are associated with a Link and are not end-to-end. Links in an MR topology may implement a different number of VLs.

Multi-Root Aware Components support one or more Virtual Hierarchies (VHs). As defined by the *PCI Express Base Specification*, each VH associated with a Port of an MRA component may support one to eight VCs. The notation (VH*x*, VC*y*) is used to denote VC *y* associated with VH *x*. Each VC of a VH associated with an MRA component Port represents an independent logical data flow that in an MR topology must be mapped to a physical VL resource in order for data to be transferred across a Link.

(VH, VC)s may be mapped to VLs in a flexible manner (e.g., (VH 0, VC 0), (VH 1, VC 2), and (VH 2, VC 1) all map to VL 0) or in a VC consistent basis (e.g., (VH*any*, VC 0) all map to VL 0). While not a requirement, systems will generally map (VH, VC) data flows with similar QoS characteristics onto the same VL.

A graphical example of mapping VCs associated with VHs to VLs is illustrated in Figure 8-1, where an MRA device connects to an MRA downstream Switch Port. In this example, the MRA device implements two VHs, each with two virtual channels. In this example, VC 0 from both VHs has been mapped to VL 0 (i.e., (A, 0) and (B, 0) both map to VL 0) while VC 1 from VH A maps to VL 1 and VC 1 from VH B maps to VL 2. The mapping of (VH, VC)s associated with a Port of an MRA component onto VLs is described in Sections 4.2.4.4 and 4.3.6.2.

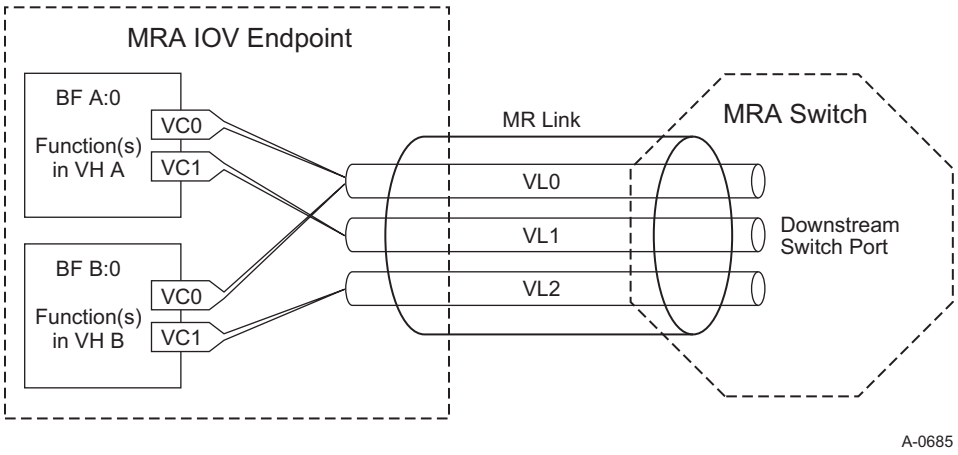

Figure 8-1: (VH, VC) to VL Mapping

8.2.1.1. Virtual Link and Virtual Hierarchy Identification

A Port of an MRA component may support from one to eight VLs. Virtual Links are uniquely identified using a Virtual Link Identification (VL ID). There is a fixed one-to-one mapping between VL IDs and VL resources (e.g., VL resource 0 always has a fixed ID of zero (i.e., VL0)).

A Port of an MRA component may support from one to 256 Virtual Hierarchies. Virtual Hierarchies associated with a Port are uniquely identified using a Virtual Hierarchy Number (VHN). VHNs are Link specific and do not represent a global VH identifier.

Each Port is independently configured and managed allowing implementations to vary the number of VLs and VHs supported per Port based usage model-specific requirements.

MR DLLPs used for flow control accounting contain VHN and VL ID information. Unlike PCIe Base TLPs that contain only TC and no VC information in the header, the MR TLP prefix tag contains both VHN and VL ID information simplifying the Flow Control accounting done at each Port of a Link.

Rules for allocating VL IDs to VL hardware resources associate with a Port are as follows:

- VL ID assignment must be one-to-one.
- The same VL ID cannot be assigned to different VL hardware resources within the same Port.
- VL ID 0 (VL0) is assigned and fixed to the default VL.

Rules for assigning VH VCs to VL hardware resources associate with a Port are as follows:

- (VH, VC) assignment must be the same (matching) for the two Ports on both sides of a Link.
- (VH0, VC0) is assigned at initialization, but not fixed, to the default VL.

MR-PCIM is responsible for configuring Ports on both sides of an MR Link in a consistent manner.

8.2.1.2. VL and VC Configuration

Support for VLs beyond the default VL0 is optional. VL0 is always enabled and while not fixed or "hardwired" by default, there is a one-to-one mapping between VC and VLs. Therefore, MR topology initialization may proceed using (VH 0, VC 0) mapped to VL 0 and does not require any specific hardware or software configuration.

MR-PCIM is responsible for enabling VLs and configuring the mapping of VC associated with VHs to VLs.

- VL0 is always enabled.
- For VLs 1-7, a VL is considered enabled when the corresponding VL Enable bit in the MR-IOV Control register has been set to 1b in the BF and once FC negotiation for that VL has exited the MRFC_INIT2_VL state.

For VLs 1-7, MR PCIM must use the VL Negotiation Pending bit in the MR-IOV Status register to determine when a VL is enabled.

Every VC resource of a VH associated with a Port visible to software operating in the VH must be mapped to an enabled VL. Since the number of VLs supported by components on a Link is implementation specific, and only one VC of any VH may be mapped to given a VL, the number of advertised VC resources to software operating in the VH must not exceed the number of enabled VLs associated with the Port.

- If a function only implements the default VC0 resource, no configuration is necessary.
- For Devices, this is managed through the Base Function as follows:
  - For Virtual Hierarchies that do not have a MFVC Capability structure associated with the Port, then the VC Extended VC Count field in the Function Control 2 register must be initialized to a value such that the number of VC resources advertised to software operating in the VH is less than or equal to the number of enabled VLs. As a result of this configuration, the VC Low Priority Extended VC Count field in the Function Control 2 register may need to be initialized to a value consistent with the VC Extended VC Count field.
  - For virtual hierarchies that have a MFVC Capability structure associated with a Port, the MFVC Extended VC Count field in the Function Control 1 register must be initialized to a value such that the number of VC resources advertised to software operating in the VH is less than or equal to the number of enabled VLs. As a result of this initialization, the MFVC Low Priority Extended VC Count field in the Function Control 1 register may need to be initialized to a value consistent with the MFVC Extended VC Count field.
- For Switches, this is managed through the corresponding Virtual Switch (VS) Bridge Table Entry as follows:
  - The VC Extended VC Count field in the Switch VS Bridge Control 2 register must be initialized to a value such that the number of VC resources advertised to software operating in the VH is less than or equal to the number of enabled VLs. As a result of this configuration, the VC Low Priority Extended VC Count in the Function Control 2 register may need to be initialized to a value consistent with the VC Extended VC Count field.

TLPs generated by a VH but not specifically associated with a BF, PF, or VF (e.g., interrupts enabled in the Device MR-IOV register) require a mapping to an associated VH. Mapping for these TLPs is performed by the Default VL field in the Device MR-IOV Control register.

TLPs generated by a BF require a mapping to an associated VH. Mapping for these TLPs is performed by the BF VL field in the Device MR-IOV Control register.

8.2.1.3. VC to VL Mapping

A Virtual Link is established when one or more VC IDs from different VHs are associated with a physical resource designated by a VL ID.

Components with Ports that implement VLs beyond the default VL must also implement an associated VC to VL mapping capability. The VC to VL mapping capability is optional for Ports that implement only the default VL0. Ports that do not have an associated VC to VL mapping capability must map VC0 from all supported VHs to VL0.

In order to preserve the semantics of a VC defined in the *PCI Express Base Specification*, only one VC of a given VH may be mapped to a VL. The behavior when two or more VCs from the same VH are mapped to a single VL is undefined.

Given the above requirement, knowledge that a VH is mapped to a VL together with the VC to VL mapping function associated with that VH is sufficient to determine the (VH, VC). Thus, indicating that a VL has a mapped VH, or (VH VL) is synonymous with specifying the (VH VC).

VC to VL mapping is from virtual VC IDs to VLs and is controlled as follows:

- For Devices, the VC to VL mapping is controlled by fields in the Function VC to VL Map register associated with the BF of each VH. This map is from Virtual Hierarchy VC resources as they would have appeared on the Link in an equivalent PCIe Base component. Thus, if function 0 in the VH contains a MFVC Capability structure, then this mapping is from VC IDs managed by the MFVC Capability structure. Otherwise, this mapping is from VC IDs managed by the BF VC capability structure.
- For Switches, the VC to VL mapping is controlled by fields in the VC to VL Map register in a VS Bridge Table entry.
- A VC ID *x* is mapped to a VL *y* when the VC*x* VL Map field has been initialized with a value of *y* and the corresponding VC*x* VL Map Enable bit has been set.

VC to VL mapping may be performed during MR-PCIM initialization or on an as-needed basis as dictated by software operating in the VH. The *PCI Express Base Specification* supports the arbitrary mapping of VC IDs to VC resources. Thus, in general, MR PCIM has no *a priori* knowledge of which VC IDs will be used or how they will be allocated by software operating in the VH. In systems where MR PCIM possess this knowledge or in which MR PCIM can communicate desired allocation to software operating in the VH, VC IDs may be mapped to VLs during MR-PCIM initialization (i.e., prior to the instantiation of software operating in the VH). Otherwise, this allocation must be performed by MR-PCIM on an as-needed basis as VC IDs are allocated by software operating in the VH to VC resources.

The VC ID to VC resource allocation and TC to VC map configuration performed by software operating in the VH may be determined by MR-PCIM via polling or interrupts as follows:

- For Devices, this information may be determined by examining the Function Table Resource Status and Function Table Multi-Function Resource Status registers. Any modification of these fields can be used to trigger a BF interrupt to MR PCIM.
- For Switches, this information may be determined by examining the VC Resource Fields register. Any modification of these fields can be used to trigger a VS Bridge interrupt to MR-PCIM.

If software operating in the VH enables a VC that has not been mapped to a VL, then the VC Negotiation Pending bit in the VC Resource Status register remains set until the VC ID has been mapped by MR-PCIM to a VL. Once mapped to a VL, the VC Negotiation Pending bit is cleared.

Once mapped, VC to VL mapping may only be modified when a VL and all associated VCs are disabled. Prior to disabling a VL, VCs mapped to that VL from all virtual hierarchies must be disabled. The behavior of disabling a VL with mapped and enabled VCs is unspecified.

Mapping a VC to a disabled VL results in the VC Negotiation Pending bit in the VC Resource Status register remaining set (i.e., the VC resource does not complete the process of negotiation due to an invalid mapping). If at some later point the disabled VL becomes enabled, the state of the VC Negotiation Pending bit is undefined.

Once enabled, MR flow control information is tracked by Receivers and Transmitters for all configured VLs and (VH, VC)s. The enabling or disabling of a VC within a VH represents a logical event that does not affect the operation of a physical MR Link or flow control information tracked on that Link. For example, (VH, VC) flow control continues to be tracked during a hot-reset of a VH.

8.2.1.4. Arbitration

The objectives of MR arbitration are to provide the following:

- Guaranteed forward progress on all supported data flows
- Differentiated service characteristics for data flows associated with a VL within an MR topology
- The ability to tune bandwidth and end-to-end latency between components in an MR topology

MRA components that support multiple VHs require an arbitration mechanism associated with each supported VL at egress Ports to select the VH from which the next TLP will be transmitted on that VL. This arbitration is referred to as VH to VL arbitration.

MRA components that support multiple VLs require an arbitration mechanism at egress Ports to select the VL from which the next TLP will be transmitted on the physical Link. This arbitration is referred to as VL to Link arbitration or just VL arbitration.

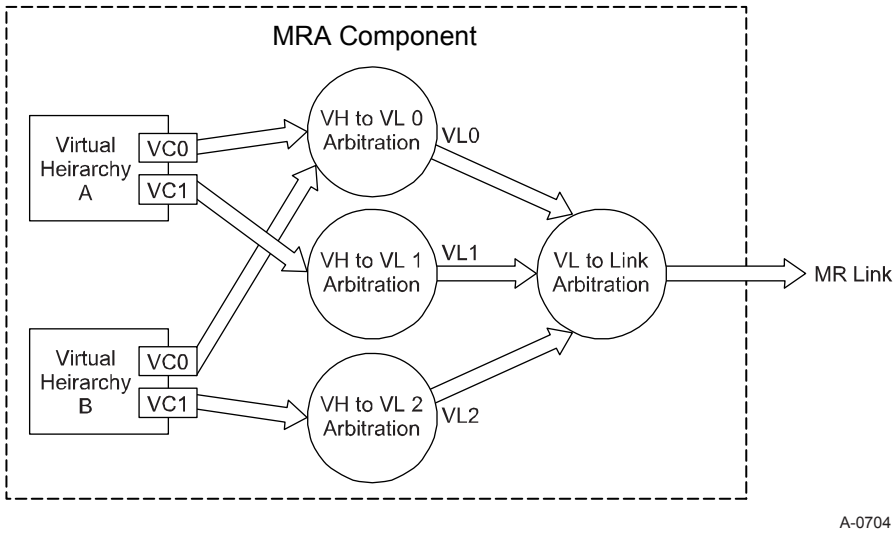

Figure 8-2: MRA Arbitration Model

Figure 8-2 illustrates both VH to VL and VL to Link arbitration associated with an egress Port of an MRA Enabled PCIe Component such as an MR Root Port, Switch, Bridge, or Device. The component in this example implements two VHs, with both VHs implementing two VCs. Flows (A, VC 0) and (B, VC 0) are mapped to VL 0 and require VH to VL arbitration to control the multiplexing of TLPs onto this VL. VLs one and two only have a single mapped flow and, therefore, require trivial arbitration. The physical Link associated with the egress Port in this example implements three VLs. VL to Link arbitration is required to control the multiplexing of VLs onto the physical Link.

8.2.1.4.1. VH to VL Arbitration

MRA components that implement multiple Virtual Hierarchies must implement a VH to VL arbitration mechanism associated with each VL supported by a Port.

VH to VL arbitration is not configurable. All implementations must support a hardwired-fixed VH to VL arbitration scheme (e.g., Round-Robin) that guarantees forward progress on all VHs associated with a VL at an egress Port.

8.2.1.4.2. VL to Link Arbitration

MRA components with Ports that support multiple Virtual Links must implement a VL to Link arbitration mechanism.

A component may support a hardwired-fixed arbitration algorithm or optional software configurable algorithms selection. Support for the optional software configurable arbitration algorithm selection is indicated by the state of the VL Arbitration Table Present bit in the MR-IOV Capabilities register.

If an implementation does not support software configurable algorithm selection, then it must implement a hardwired-fixed arbitration scheme (e.g., Round Robin) that guarantees forward progress on all enabled VLs.

The remainder of this section describes the behavior and requirements of software configurable VL arbitration algorithm selection.

VL arbitration algorithm selection is controlled as follows:

- For Devices, registers associated with VL arbitration are located in the MR-IOV Capability.
- For Switches, registers associated with VL arbitration are located in the Port Table Entry of the corresponding Port.
- The VL arbitration table in all components, if present, is located in BAR Memory Space.

VLs may be partitioned into two priority groups: a lower and an upper group. VLs in the upper group are arbitrated using strict priority based on VL number while VLs in the lower group are arbitrated only when there are no packets to process in the upper group. Arbitration within the lower group may be configured to one of the supported arbitration algorithms described below.

Membership of a VL in the low or high priority group is determined by the state of the corresponding bit in the VL Strict Priority Arbitration field in the VL Arbitration Control register. Since the VL Strict Priority Arbitration field represents a bit vector, VLs to group assignment is flexible and need not be allocated sequentially based on VL ID.

Among VLs configured for strict priority, priority is based on increasing VL number. VL0 has the lowest priority while VL 7 has the highest.

The arbitration algorithm for VLs in the low priority group is selected by the VL Arbitration Select field in the VL Arbitration Control register. Arbitration algorithms supported by an implementation are advertised in the VL Arbitration Capability field in the VL Arbitration Capability and Status register and may include the following architected schemes:

- Hardware-fixed arbitration, e.g., Round-Robin
- Weighted Round Robin (WRR) arbitration scheme with 32, 64, 128, or 256 phases
- Time-Based Weighted Round Robin (time-based WRR) arbitration scheme with 128 phases
- Vendor defined arbitration

This specification establishes a standard framework within which vendors may specify their own vendor specific arbitration scheme. The definition of vendor-defined arbitration is outside the scope of this document.

VL arbitration algorithms, e.g., WRR and time-based WRR, operate in a manner analogous to the schemes defined for VC arbitration in the *PCI Express Base Specification*.

8.2.2. Bypass Queues

Associated with each VL at a receiver are dedicated physical resources (queues/buffers and control logic) that allow independent traffic flows to proceed on the Port. This section describes the logical queuing structure associated with each VL at a Receiver. This logical description does not imply or require a particular implementation and is used solely to clarify requirements.

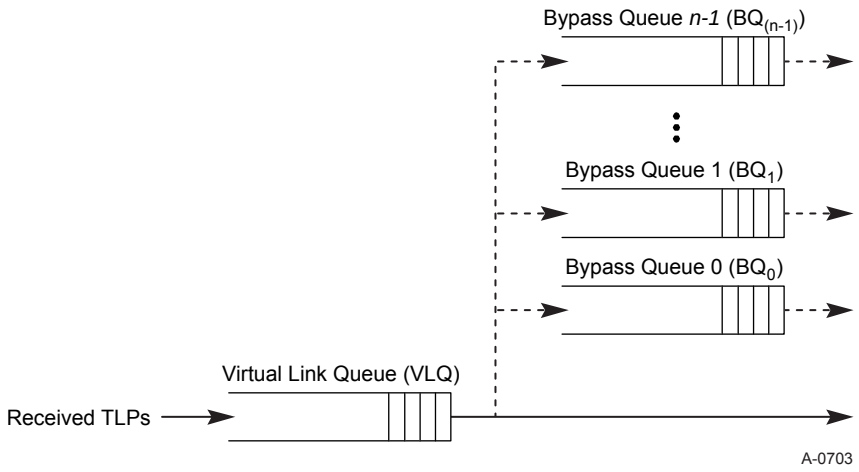

Figure 8-3: Logical Queuing Structure Associated with a VL Receiver

As illustrated in Figure 8-3, associated with each VL at a receiver is a Virtual Link Queue (VLQ) and one or more optional Bypass Queues (BQ). Within a VL, the PCIe ordering rules defined in the *PCI Express Base Specification* are maintained with respect to TLPs from a VH across both the VLQ and the associated BQ (if any). TLPs associated with a VL received on a Link are queued in the VLQ. In the absence of congestion, TLPs for non-congested VHs are dequeued from the VLQ for receiver processing.

When a TLP at the head of the VLQ experiences congestion, a free BQ is allocated to the VH associated with the TLP, if one does not already exist, and the TLP is moved to the BQ. Subsequent TLPs associated with the VH from the VLQ are moved to the same BQ until the congestion condition clears and the (now empty) BQ is freed. This scheme allows TLPs associated with a congested flow to be bypassed and isolates congestion to a VH. TLPs are dequeued from a BQ as flow control credits of the required type for the VH associated with the BQ become available. An implementation must ensure that forward progress is guaranteed on all flows.

If at any point the number of congested VHs associated with a VL exceeds the number of BQs implemented by a Receiver for that VL, then TLPs in the VLQ experience congestion. Thus, congestion is managed within a VL at the receiver by dynamically mapping congested VH flows onto BQs. The degree of congestion isolation between flows mapped to the same VL is dictated by the number of BQs implemented for that VL by the receiver. A receiver that implements only the required VLQ provides no congestion isolation between data flows mapped to the VL, while a receiver that implements *n* optional BQs provides complete congestion isolation for up to *n* flows.

8.2.3. Flow Control Rules

- Components must implement independent Flow Control of all supported VLs.
- As in the *PCI Express Base Specification*, flow control is distinguished between TLP type (Posted, Non-Posted, and Completion) and Header/Data. Thus, there are six types of tracked flow control information.
- The unit of Flow Control credit is 4 DW for data.

- The unit of Flow Control credit for headers is one maximum-size header plus TLP prefix and TLP digest.
- Flow Control is initialized autonomously by hardware only for the default virtual Link (VL0) and virtual hierarchy (VH0).
- When other Virtual Links are enabled by MR-PCIM, each newly created VL will follow the flow control initialization protocol.
- (VH VL) credits are negotiated using the flow control initialization protocol outlined in Section 2.1.2 whenever MR-PCIM increases NumVH.
- A Receiver must never cumulatively issue more than 2047 outstanding unused credits to the Transmitter for data and 127 for header.
- If an Infinite VL and (VH VL) credit advertisement has been made during initialization, no Flow Control updates are required for that VL following initialization.
- A Receiver that advertises non-infinite VH credits must utilize MRUpdateFC DLLPs for that VL.
  - Independent MRUpdateFCs DLLPs are used to track header and data credits associated with VLs and (VH VL)s.
  - As described in Sections 2.4.1 and 2.4.2, Receivers and Transmitters track independent flow control information for each VL and for each supported VH. For each VL and (VH VL), the six types of flow control information outlined above are tracked.
  - A TLP in a Receiver's VLQ or BQ consumes both VL and corresponding VH credits.
  - Both VL and corresponding VH credits are released when a TLP is processed and removed from the logical queuing structure associated with the Receiver for a VL.
  - MRUpdateFC DLLPs are only associated with VH credits related with a VL. VL credits are implicitly computed using state information and MRUpdateFC DLLPs as outlined in Section 2.4.1.
- A Receiver that advertises non-infinite VL credits and infinite VH credits must utilize PCIe Base UpdateFC DLLPs for that VL.
  - UpdateFC DLLPs are used to explicitly track VL header and data credits.
  - The VH gating function is unconditionally satisfied for all Credit Types associated with that VL.
  - Receivers and Transmitters track independent flow control information for each VL. For each VL, the six types of flow control information are tracked.
  - A Receiver that advertises infinite VH credits may only implement a VLQ for that VL.
  - VL credits are released when a TLP is processed and removed from the logical queuing structure associated with the Receiver for a VL
- If a receiver advertises infinite VH credits in a given VL, it must advertise infinite VH credits on all VHs in that VL.

8.3. Performance Monitoring and Statistics Collection

Congestion in a Multi-Root Topology may result in traffic associated with one Virtual Hierarchy affecting the performance of an unrelated Virtual Hierarchy. This section defines a set of optional performance monitoring and statistics collection capabilities that may be used to diagnose, plan, and tune the performance of a multi-root system.

The objective of standardizing these capabilities is to allow component vendor independent software to monitor performance and manage congestion. It is not a goal of these capabilities to monitor or count errors.

This capability is optional for all MR-IOV components, but its implementation is strongly encouraged for all MRA Switch Ports.

Registers and tables associated with the Performance Monitoring and Statistics Collection capability are described in Section 4.5. This section describes functional behavior.

A component that implements the optional Performance Monitoring and Statistics Collection Capability is required to implement basic features outlined in this section and in Section 4.5. These features ensure minimum functionality and interoperability with software that utilizes these capabilities.

Support for the Performance Monitoring and Statistics Collection Capability is indicated by a non-zero value in the Statistics Capability register location in the MR-IOV Extended Capability.

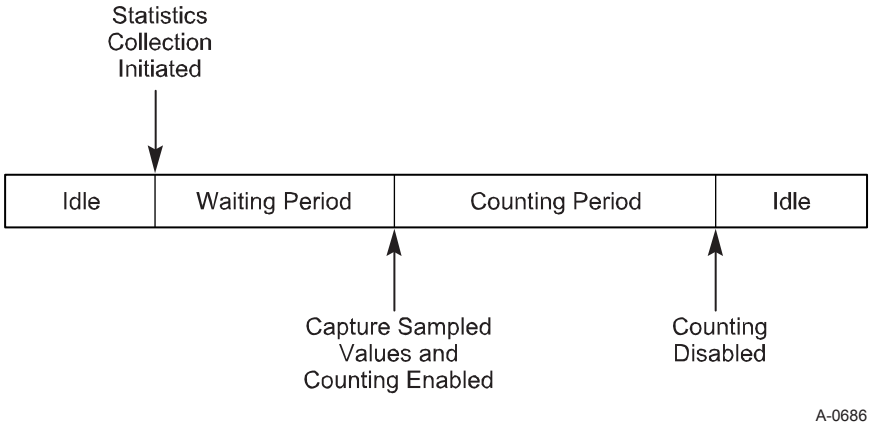

Figure 8-4: Statistics Collection Process

Performance statistics are recorded by Statistics Counters and may be captured values and counted values. Captured values correspond to sampled system state while counted values correspond to the number of occurrences of a selected event over a counting period. The statistics collection process is illustrated in Figure 8-4.

Completion of the statistics collection process (i.e., the end of the counting period) may be signaled via an interrupt.

A component that implements the Performance Monitoring and Statistics Collection Capability must implement a Statistics Block Table and a Statistics Descriptor Table. These tables are located in Memory Space and their location is specified by the Statistics Descriptor Table and Statistics Block Table registers in the MR-IOV Extended Capability structure.

A set of Statistics Counters that share a common initiation mechanism and statistics collection process periods is referred to as a Statistics Block. A component that implements the Performance Monitoring and Statistics Collection Capability may implement one to 32 Statistics Blocks. Each Statistics Block has an associated Statistics Block Table entry that contains a pointer to a Statistics Counter Table that holds the Statistics Counters associated with the Statistics Block. The Statistics Block Table entry also specifies the statistics collection process state (i.e., Idle, Waiting, Counting), number of entries in the Statistics Counter Table, waiting period, and counting period.

Statistics Counters associated with a Statistics Block may have different characteristics and be associated with different Ports. Associated with each Statistics Counter is a Statistics Descriptor Index that points to the Statistics Descriptor Table entry that describes statistics that may be recorded by the Statistics Counter. The actual statistic recorded by a Statistics Counter is selected by the Statistics Select field.

Associated with a Statistics Counters is a 64-bit counter that is used to report the captured statistic. The counter is formed by the Statistics Counter Low and Statistics Counter High registers. While the field is specified as 64 bits, an implementation is free to implement fewer bits. The number of implemented counter bits is specified by the Statistics Width field and, for standard counters, must be 32 bits or greater.

Associated with each Statistics Counter is an optional filter specified by the Statistics Filter Enable and Control register. Filters allow refinement in a recorded statistic. For example, rather than count all transmitted TLPs on a Port, a filter may be used to only count transmitted TLPs from a particular VH on a particular VL. Each entry in a Statistics Descriptor (i.e., an S bit) defines required filters that must be implemented and optional filters that may be implemented.

The format of Statistics Descriptors, standard statistics, filters, and requirements are specified in Section 4.5.2.

A component that implements the Performance Monitoring and Statistics Collection Capability must implement at least one Statistics Block and at least two Statistics Counters per Port. At least two Statistics Counters per Port must implement Statistics Descriptor standard statistics specified as required in Table 4-72.

Acknowledgements

The following people were instrumental in the development of the MR-IOV Specification[17].

Shawn Clayton, Emulex Corporation

Eric DeHaemer, Intel Corporation

Robert Dickson, Sun Microsystems, Inc.

Jeff Fose, Emulex Corporation

Douglas Freimuth, IBM Corporation

Steve Glaser, NextIO, Inc.

Lucian Gozu, Neterion Corporation

Andrew Gruber, Advanced Micro Devices, Inc.

Mark Hummel, Advanced Micro Devices, Inc.

David Kahn, Sun Microsystems, Inc.

Kwok Kong, IDT Corporation

Michael Krause, Hewlett-Packard Company

Brian Langendorf, Nvidia Corporation

Paul Mattos, IBM Corporation

David Mayhew, Advanced Micro Devices, Inc.

Richard Moore, Qlogic Corporation

Peter Onufyrk, IDT Corporation

Jake Oshins, Microsoft Corporation

Chris Pettey, NextIO, Inc.

Renato Recio, IBM Corporation

Jack Regula, PLX Corporation

Wesley Shao, Sun Microsystems, Inc.

Richard Solomon, LSI Corporation

Steven Thurber, IBM Corporation

Robert Utley, NextIO Inc.

Mahesh Wagh, Intel Corporation

Jim Williams, Emulex Corporation

[17] Company affiliation is at the time of specification contribution.

Theodore Willke, Intel Corporation

David Wooten, Microsoft Corporation

William Wu, Broadcom Corporation